NASA Contractor Report

A Computer Code for Multiphase All-Speed Transient Flows in Complex Geometries

MAST version 1.0

C. P. Chen *
Y. Jiang
Y. M. Kim
H. M. Shang

University of Alabama in Huntsville
Huntsville, Alabama 35899.

Prepared for
Marshall Space Flight Center
Under Grant NAG8-092

* Authors listed alphabetically

October 1991

TABLE OF CONTENTS

SUMMARY

The operation of the MAST code, which computes transient solutions to the multiphase flow equations applicable to all-speed flows, is described. Two-phase flows are formulated based on the Eulerian-Lagrangian scheme in which the continuous phase is described by the Navier-Stokes equation (or Reynolds equations for turbulent flows). Dispersed phase is formulated by a Lagrangian tracking scheme. The numerical solution algorithms utilized for fluid flows is a newly developed Pressure-Implicit algorithm based on the operator-splitting technique in generalized non-orthogonal coordinates. This operator split allows separate operations on each of the variable fields to handle pressure-velocity coupling. The pressure correction equation thus obtained has the hyperbolic nature and is effective for Mach numbers ranging form the incompressible limit to supersonic flow regimes. The present code adopts a non-staggered grid arrangement, thus the velocity components and other dependent variables are collocated at the same grid. A sequence of benchmark-quality problems, including incompressible, subsonic, transonic, supersonic, gas-droplet two-phase flows, as well as spray-combustion problems, were performed to demonstrate the robustness and accuracy of the present code.

CHAPTER 1

INTRODUCTION

The partial differential equations governing fluid flows in a typical liquid-fueled rocket engine are nonlinear and are too complex to obtain the solution in analytical form in general, therefore computational methods for simulating flow fields must be employed. Since problems of practical interest include a very wide range of flow situations, it would be desirable to develop a single numerical algorithm capable of reliably handling many flow conditions in an efficient manner.

The numerical modeling of the physical mechanisms of turbulence effects, multiphase flows, and combustion are very complex individually and the formulation of an algorithm combining all of the above mechanisms would be a formidable task. However, an approach can be adopted such that one aspect of the numerical simulation can be considered individually and collectively common to the above process; this aspect being the transient all-speed Navier-Stokes flow solver. As the first logical step in attaining a fully comprehensive chemical reacting flow simulator, we develop a versatile and generalized flow solver which could efficiently handle multidimensional, incompressible and compressible, laminar and turbulent, reacting and non-reacting, steady and unsteady, single- and multi-phase flows. This report is intended as reference guide for users of the MAST (Multiphase All-Speed Transient) family codes.

The primary features of the MAST code can be summarized as follows:

1. A time-accurate formulation is employed, thus a steady state solution is achieved through a temporal marching approach.

2. A pressure-based method using pressure and velocity as primary variables is formulated on a non-staggered, general, non-orthogonal coordinate system.

3. The pressure-velocity coupling is handled through a modified, non-iterative operator-splitting algorithm valid for all speed flow situations.

4. The set of discretized equations is solved by using a conjugate gradient (CGS) method.

5. The dispersed phase is handled through a Lagrangian tracking technique embedded in the pressure-velocity predictor-corrector algorithm for efficient two-way coupling.

In the following, fundamental elements for establishing the MAST code will be described. These are followed by a series of bench-mark fluid dynamics calculations. A user's guide is also provided in Appendixes.

CHAPTER 2

GOVERNING EQUATIONS

The unsteady, Newtonian, viscous compressible flow is governed by the following Navier-Stokes equations:

$$\frac{\partial}{\partial t}(\rho)+\frac{\partial}{\partial x_i}(\rho u_i)=0. \tag{1}$$

$$\frac{\partial}{\partial t}(\rho u_i)+\frac{\partial}{\partial x_j}\left(\rho u_j u_i\right)=-\frac{\partial p}{\partial x_i}+\frac{\partial}{\partial x_j}\sigma_{ij}+S_i \tag{2}$$

$$\frac{\partial}{\partial t}(\rho h)+\frac{\partial}{\partial x_i}(\rho u_i h)=\frac{\partial p}{\partial t}+\frac{\partial}{\partial x_i}\left(\frac{\mu}{\Pr}\frac{\partial h}{\partial x_i}\right)+\frac{\partial}{\partial x_i}\left[\left(1-\frac{1}{\Pr}\right)\mu\frac{\partial\frac{1}{2}u_j u_j}{\partial x_i}\right]+Q \tag{3}$$

In the above ρ is the density, u_i the velocity components, p the pressure, S_i are body forces and σ_{ij} are the components of deviatory stress tensor. In equation (3), h is the stagnation enthalpy of the fluid, and is related to the specific energy e (which is a function of temperature T) by

$$h=e(T)+\frac{p}{\rho}+\frac{1}{2}u_i u_i$$

The viscous constitutive relation

$$\sigma_{ij}=\mu\left[\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}-\frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right] \tag{4}$$

and the density distributions are provided by the equation of state.

$$\rho=\frac{p}{RT} \tag{5}$$

To treat arbitrary boundaries, these governing equations are transformed to a general form based on a nonorthogonal coordinate system (ξ,η,ζ). There are three ways to represent velocity field on a nonorthogonal grid system; i.e. (1) Cartesian Velocity Components [1,2], (2) Contravariant velocity component [3] and (3) resolute or velocity projections along the coordinate directions [4,5]. The present approach uses the Cartesian components of the velocity vector stored at grid points. This has the advantage of excluding the so called extra "curvature" terms in the momentum equations. Using the standard transformations formula [6], the continuity equation becomes

$$\frac{\partial \rho}{\partial t}+\frac{1}{J}\left[\frac{\partial}{\partial \xi}(\rho U)+\frac{\partial}{\partial \eta}(\rho V)+\frac{\partial}{\partial \varsigma}(\rho W)\right]=0 \qquad (6)$$

and the remaining governing equations for the dependent variables ϕ in the general coordinate system can be expressed in a compact form as follows:

$$\begin{aligned}&\frac{\partial}{\partial t}(\rho\phi)+\frac{1}{J}\left[\frac{\partial}{\partial \xi}(\rho U\phi)+\frac{\partial}{\partial \eta}(\rho V\phi)+\frac{\partial}{\partial \varsigma}(\rho W\phi)\right]\\ &=\frac{\partial}{\partial \xi}\left[\frac{\Gamma_\phi}{J}\left(g_{11}^2+g_{12}^2+g_{13}^2\right)\frac{\partial \phi}{\partial \xi}\right]+\frac{\partial}{\partial \eta}\left[\frac{\Gamma_\phi}{J}\left(g_{21}^2+g_{22}^2+g_{23}^2\right)\frac{\partial \phi}{\partial \eta}\right]+\\ &\frac{\partial}{\partial \varsigma}\left[\frac{\Gamma_\phi}{J}\left(g_{31}^2+g_{32}^2+g_{33}^2\right)\frac{\partial \phi}{\partial \varsigma}\right]+S^\phi \qquad (7)\end{aligned}$$

where U, V, W are the contravariant variables that represent convective fluxes;

$$U = g_{11}u+g_{12}v+g_{13}w$$

$$V = g_{21}u+g_{22}v+g_{23}w$$

$$W = g_{31}u+g_{32}v+g_{33}w$$

or

$$U_i = g_{ij}u_j \qquad (8)$$

and Γ_ϕ and S^ϕ are the associated diffusivity and source terms for the variable ϕ (= u, v, w, T, κ, ε, etc.). The detailed expressions of the source terms and the cross derivative terms due to grid non-orthogonality are listed in Table 1. The geometric coefficient g_{ij} and the Jacobian J are defined as

$$g_{ij} = \begin{pmatrix} y_\eta z_\varsigma - y_\varsigma z_\eta & x_\varsigma z_\eta - x_\eta z_\varsigma & x_\eta y_\varsigma - x_\varsigma y_\eta \\ y_\varsigma z_\xi - y_\xi z_\varsigma & x_\xi z_\varsigma - x_\varsigma z_\xi & x_\varsigma y_\xi - x_\xi y_\varsigma \\ y_\xi z_\eta - y_\eta z_\xi & x_\eta z_\xi - x_\xi z_\eta & x_\xi y_\eta - x_\eta y_\xi \end{pmatrix} \qquad (9)$$

and

$$J = x_\xi y_\eta z_\varsigma + x_\varsigma y_\xi z_\eta + x_\eta y_\varsigma z_\xi - x_\xi y_\varsigma z_\eta - x_\eta y_\xi z_\varsigma - x_\varsigma y_\eta z_\xi \qquad (10)$$

For turbulent flow computations, the MAST code has employed the two-equation κ–ε model with wall-functions. In this case, the molecular viscosity and thermal diffusivities are augmented by the effective viscosity and effective diffusivities as:

$$\mu_{eff} = \mu + \mu_t \qquad (11)$$

$$D_{eff} = \frac{\mu}{Pr} + \frac{\mu_t}{P_{rt}} \qquad (12)$$

in which

$$\frac{\mu_t}{\rho} = C_\mu \frac{\kappa^2}{\varepsilon} f_\mu \qquad (13)$$

The transport equations for the κ and ε can also be cast in the standard form with source terms listed in Table 2.

COMPUTATIONAL METHODS

A. Discretization

Discretization of the governing, differential equations uses the finite volume approach [7]. Differencing in the temporal domain employs the Implicit Euler scheme. All the dependent and independent variables are stored at the same grid location and variables at the finite control volume boundaries are interpolated between adjacent grid points. In the spatial domain, the diffusion terms are approximated by the central differencing scheme. To handle sharp shock-capturing without oscillatory fields near the shock discontinuity, a high accuracy scheme for handling the convection terms is implemented. This scheme is briefly described for the convective flux term in the x direction. Let the discretized formulation of the invicid component for grid point be

$$\frac{\partial(\rho U \phi)}{\partial x}\Big|_i = \frac{\partial f}{\partial x}\Big|_i \cong \frac{f_{\,i+\frac{1}{2}} - f_{\,i-\frac{1}{2}}}{\Delta x} \qquad (14)$$

then the control-volume interface flux is computed according to:

$$f_{i+\frac{1}{2}} = h_{i+\frac{1}{2}} + \frac{(1+\Phi)}{4} d\bar{f}^{+}{}_{i+\frac{1}{2}} - \frac{(1+\Phi)}{4} d\bar{f}^{-}{}_{i+\frac{1}{2}}$$

$$+ \frac{(1-\Phi)}{4} d\bar{f}^{+}{}_{i-\frac{1}{2}} - \frac{(1-\Phi)}{4} d\bar{f}^{-}{}_{i+\frac{3}{2}} \qquad (15)$$

In the above, $h_{i+\frac{1}{2}}$ represents a first - order numerical flux, and can be computed from:

$$h_{i+\frac{1}{2}} = \rho_{i+\frac{1}{2}} U^{+}_{i+\frac{1}{2}} \phi_i + \rho_{i+\frac{1}{2}} U^{-}_{i+\frac{1}{2}} \phi_{i+1} \qquad (16)$$

where $$U^{\pm}_{i+\frac{1}{2}} = 0.5 \times \left(U_{i+\frac{1}{2}} \pm \left| U_{i+\frac{1}{2}} \right| \right)$$

The flux-limited values of df are computed as follows:

$$df^{+}{}_{i+\frac{1}{2}} = \rho_{i+\frac{1}{2}} U^{+}_{i+\frac{1}{2}} (\phi_{i+1} - \phi_i) \qquad (17.a)$$

$$df^{+}{}_{i-\frac{1}{2}} = \rho_{i+\frac{1}{2}} U^{+}_{i+\frac{1}{2}} (\phi_i - \phi_{i-1}) \qquad (17.b)$$

$$df^{-}{}_{i+\frac{1}{2}} = \rho_{i+\frac{1}{2}} U^{-}_{i+\frac{1}{2}} (\phi_{i+1} - \phi_i) \qquad (17.c)$$

$$df^{-}{}_{i+\frac{3}{2}} = \rho_{i+\frac{1}{2}} U^{-}_{i+\frac{1}{2}} (\phi_{i+2} - \phi_{i+1}) \qquad (17.d)$$

In contrast to the minimod operator used in [8] , we use the following "compression" parameters determined by a local normalized pressure gradient monitor as follows:

$$d\bar{f}^{\pm}_{i+\frac{1}{2}} = \tilde{\beta}_{i+\frac{1}{2}} df^{\pm}_{i+\frac{1}{2}} \qquad (18.a)$$

$$d\bar{f}^{+}_{i-\frac{1}{2}} = \tilde{\beta}_{i-\frac{1}{2}} df^{+}_{i-\frac{1}{2}} \qquad (18.b)$$

$$d\bar{f}^{-}_{i+\frac{3}{2}} = \tilde{\beta}_{i+\frac{3}{2}} df^{-}_{i+\frac{3}{2}} \qquad (18.c)$$

where $\tilde{\beta}_{i+\frac{1}{2}} = \min(g_i \ , g_{i+1})$

where $$g_i = \begin{cases} 0 & ; \ \alpha_i < 0 \\ \dfrac{(A + B\alpha_i)}{A\alpha_i + B} & ; \ 0 \le \alpha_i < 1 \\ \dfrac{A + B}{A + B\alpha_i} & ; \ \alpha_i \ge 1 \end{cases}$$

and $\alpha_i = \dfrac{p_{i+1} - p_i}{p_i - p_{i-1}}$

In this study, A and B are set to be 1 and Φ is chosen to have high resolution for specific flow regimes. Suggested values are given in the results sections.

B. Velocity-Pressure Coupling

A typical computational procedure for compressible flows employs density as a primary variable and extracts pressure from an equation of state. This density-based method has difficulties in handling incompressible or low-Mach number flows, due to weak pressure-density coupling.

Computation for incompressible flow is generally performed using a pressure based method which is characterized by the use of pressure as one of the primary dependent variables. However, the absence of an explicit equation for pressure imposes that the pressure can only influence the velocity field through the momentum conservation with the continuity as a compatibility condition. Hence provision of algorithm for the velocity-pressure coupling is essential. There are several algorithms employing a semi-implicit iterative coupling procedure; e.g. SIMPLE [7] and its variants [9]. In the MAST code, an efficient non-iterative method based on the operator-splitting technique is extended to non-orthogonal grid coordinate. The discretized governing equations are solved in a time-marching fashion. In each time step, a fixed number of correction stages follow an initial predictor stage, and this predictor-multiple corrector procedure yields a time-accurate result.

As is the case with the basic operator-splitting method depicted in [10], we seek to discretize the governing equations as the following by using the simple Euler implicit scheme in temporal domain:

$$\frac{1}{\delta t}\left\{(\rho u_i)^{n+1} - (\rho u_i)^{n}\right\} = H'\left(u_i^{n+1}\right) - \Delta_i p^{n+1} + S_i \tag{19}$$

$$\frac{1}{\delta t}\left\{(\rho h)^{n+1} - (\rho h)^{n}\right\} = J'\left(h^{n+1}\right) + \frac{p^{n+1} - p^{n}}{\delta t} + Q_i \tag{20}$$

In Eq. (19) and (20), finite difference operator H'( ) and J' ( ) contains the convection and diffusion operations. They can be further split into:

$$H'\left(u_i^{n+1}\right) = A_0 u_i^{(\;)} + H\left(u_i^{(\;)}\right) \tag{21.a}$$

$$J'\left(h^{n+1}\right) = B_0 h^{(\;)} + J\left(h^{(\;)}\right) \tag{21.b}$$

in which A_o and B_o denote the finite difference coefficient matrix for diagonal elements. The splitting of operator in (21.a) and (21.b) allows different intermediate time steps be used. This technique was introduced to derive the velocity-pressure coupling sequence in several intermediate steps. Let superscripts *, ** and *** represent predictor, first and second corrector field values obtained during the splitting procedure. For simplicity, only single-phase, laminar pressure-velocity coupling equations are described here first.

(a) Predictor step. The current velocity, density and pressure field are used to calculate finite-volume discretization coefficient to solve the implicit momentum predictor step:

$$\left(\frac{\rho^{n}}{\delta t} - A_0\right) u_i^{*} = H\left(u_i^{*}\right) - \Delta_i p^{n} + S_i + \frac{\rho^{n-1} u_i^{n}}{\delta t} \tag{22}$$

In this step, $H'(u^*_i) = A_0 u^*_i + H(u^*_i)$

(b) First momentum corrector step. To derive a pressure-velocity coupling formulation satisfying conservation of mass is the goal of this step. The form of pressure equation depends greatly on the discretization of the continuity equation. In this study, we propose the following form of the continuity equation:

$$\frac{\rho^* - \rho^{\,n}}{\delta t} + \Delta_i\left(\rho^n u_i^{**}\right) + \Delta_i\left(\rho' u_i^*\right) = 0 \qquad (23)$$

where $\rho' = \rho^* - \rho^n$

This equation differs from the original PISO formulation and is consistent with the linearization procedure used in momentum equations as well as other transport equations To utilize this equation, the momentum equation (2) can be discretized in an explicit form for the second corrector form as:

$$\left(\frac{\rho^{\,n}}{\delta t} - A_0\right) u_i^{**} = H'(u_i^*) - \Delta_i p^* + S_i + \frac{\rho^{n+1} u_i^n}{\delta t} \qquad (24)$$

Note that in this step, $H'(u_i^{**}) \cong A_0 u_i^{**} + H(u_i^*)$

We now subtract Eq.(22) from Eq. (24) and then take the divergence of the resulting equation and substitute into Eq. (23), which gives the implicit pressure correction equation for the first corrector step:

$$\left\{\frac{1}{\delta t R T^n} + \Delta_i \frac{u_i^*}{R T^n} - \Delta_i\left(\rho^n D_i \Delta_i\right)\right\}\left(p^* \text{-} p^n\right) = -\Delta_i\left(\rho^n u_i^*\right) \qquad (25)$$

where $D_i = \left(\frac{\rho^n}{\delta t} - A_0\right)^{-1}$

Note that this equation involves a convection term which enhances the convergence and properly takes into account the hyperbolic nature of supersonic flows, and enables capturing shock waves. In Eq. (22), ρ^* has been substituted by the equation of state

$$\rho^* = \frac{p^*}{R T^n}$$

The energy field for the first correction step is then solved as:

$$\left(\frac{\rho^n}{\delta t} - B_0\right)h* = J(h*) + \frac{\rho^{n-1}}{\delta t}h^n + \frac{p* - p^n}{\delta t} + Q_i* \tag{26}$$

(c) Second corrector step. The first corrector step satisfies the continuity equation (23). To satisfy momentum and other equations further, a second pressure correction equation was derived using an explicit second momentum corrector and a discretized continuity equation:

$$\left(\frac{\rho^n}{\delta t} - A_0\right)u_i^{***} = H\left(u_i^{**}\right) - \Delta_i p^{**} + S_i + \frac{\rho^{n-1} u_i^n}{\delta t} \tag{27}$$

$$\frac{\rho^{**} - \rho^n}{\delta t} + \Delta_i\left(\rho^* u_i^{***}\right) + \Delta_i\left(\rho'' u_i^{**}\right) = 0 \tag{28}$$

where $\rho'' = \rho** - \rho*$

The resulting form of the second pressure correction equation becomes:

$$\begin{aligned}&\left\{\frac{1}{\delta t RT^*} + \Delta_i \frac{u_i^{**}}{RT^*} - \Delta_i\left(\rho^* D_i \Delta_i\right)\right\}\left(p^{**} - p^*\right)\\ &= -\Delta_i\left\{H\left(u_i^{**} - u_i^*\right)\right\} + \frac{p^*}{\delta t}\left(\frac{1}{RT^n} - \frac{1}{RT^*}\right) + \Delta_i\left[\rho*\left(\frac{1}{RT^n} - \frac{1}{RT*}\right)u_i**\right]\end{aligned} \tag{29}$$

Here, $\rho^{**} = \dfrac{p^{**}}{RT^*}$

was used for deriving Eq. (29). This second pressure-correction equation also differs from the PISO algorithm by a convection term on the left hand side and an extra term on the right hand side. The energy equations are solved again to obtain h** and eventually T** . It is shown in [11] that the errors introduced by the splitting procedure in the temporal domain vanish with δt and do so at least as fast as the temporal discretization error introduced by the Euler discretization scheme. The variables solved after the second corrector steps are then taken to be the field values at the next time level, n+1. It should be noted that, for incompressible flows for which ρ is a constant, the present method reduces to the original PISO algorithm [10].

C. The Collocated Grid Arrangement

It is known that the non-staggered grid system can produce undesirable checkerboard (oscillatory) phenomena in the pressure field, due to the decoupling effect between velocities and pressures. As suggested by Rhie [1], an explicit fourth order pressure damping term may be added to the right hand side of pressure increment equations to damp out pressure oscillations. This approach can be further modified to include the transient term for a time-marching formulation. However, the steady state solution obtained in this way is found to be strongly influenced by the size of time step chosen, which is undesirable [26].

The pressure-velocity coupling procedure described above is general and is not related to any particular grid arrangements and spatial discretization schemes. In this study, this coupling algorithm has been formulated in a general curvilinear non-orthogonal body-fitted coordinate system with a non-staggered grid arrangement, i.e. ,all variables are located at the same grid point. A new weighted interpolation method was developed to avoid checkerboard pressure field frequently encountered in pressure-based method utilizing non-staggered grid. This treatment is briefly described here. More detailed justifications can be found in [11]. For simplicity, let the finite difference operator acting on the u velocity component (in the x direction) at grid points i and i+1 be written as:

$$A_i u_i = H_i - \frac{\partial p}{\partial x}\Big|_i + S_i + \frac{(\rho u)_i^n}{\delta t} \tag{30}$$

$$A_{i+1} u_{i+1} = H_{i+1} - \frac{\partial p}{\partial x}\Big|_{i+1} + S_{i+1} + \frac{(\rho u)^n{}_{i+1}}{\delta t} \tag{31}$$

In the control-volume based finite difference method used in this study, the main concern is the estimation of control-volume face value u. In the adaptive interpolation method developed in [11], the face value is evaluated according to:

$$u_{i+\frac{1}{2}} = \beta_{i+\frac{1}{2}} \bar{u}_{i+\frac{1}{2}} + \left(1 - \beta_{i+\frac{1}{2}}\right) \bar{\bar{u}}_{i+\frac{1}{2}} \tag{32}$$

where $\beta_{i+\frac{1}{2}} = 1 - \max(g_{i+1}\ ,\ g_i)$

in which $g_i = \left| \frac{p_{i+1} + p_{i-1} - 2p_i}{p_{i+1} + p_{i-1} + 2p_i} \right|$

In Eq. (32), $\bar{\bar{u}}_{i+1/2}$ is simply an arithmetic averaging , $=(u_i + u_{i+1})/2$, and $\bar{u}_{i+1/2}$ is estimated by the following approximate operator:

$$(A_i + A_{i+1})\bar{u}_{i+\frac{1}{2}} = (A_{i+1} - A_i)(u_{i+1} - u_i) + H_i + H_{i+1}$$

$$-2\left.\frac{\partial p}{\partial x}\right|_{i+\frac{1}{2}} + S_i + S_{i+1} + \frac{2(\rho u)^n_{i+\frac{1}{2}}}{\delta t} \qquad (33)$$

Detailed testings of this method and the comparisons with other non-staggered treatments [12,13] can be found in [11].

D. Equation Solver

The system of finite difference approximation equations formed by equations (22), (25), (26) and (29) produces a nine-banded matrix. To enhance the matrix solver efficiency, the conjugate-gradient-squared (CGS) method as described in [14] is employed.

DISCRETE PARTICLE TRACKING METHOD

In the MAST code, the dispersed phase, i.e. the spray, is considered to be composed of discrete computational parcels, each of which represents a group of droplets of similar size, velocity, temperature etc. The distribution function in droplet size, velocity, temperature, spray pattern produced by the fuel injector (or atomizer) is statistically sampled and the resulting computational parcels are followed as they interact and exchange moment, energy and mass with the surrounding gas. The present numerical model employs this Monte Carlo method and the basic ideas of implementing this technique in the present CFD method are presented in Ref [15], which is reproduced in Appendix C. Detailed descriptions of the sampling technique of specified distribution functions for characterizing the initial conditions for spray and the subsequent droplet-fluid interactions can also be found in Ref [15] and [16].

One of the important aspects in spray combustion modeling is the dense spray effects which include atomization process, drop breakup, droplet collision and coalescence. Atomization process occurs on time and length scale too short to be resolves with practical computation grid sizes and time steps. Thus, atomization should be modeled as a sub-grid-scale process. To account for the dense spray effects, the present study employs the drop collision and coalescence model [33] and the Taylor analogy breakup (TAB) model [34]. In the drop collision model, the probability distributions governing the number and outcomes of the collisions between two drops are sampled randomly in consistency with the stochastic particle tracking method. The TAB model utilizes an analogy between an oscillating and distorting droplet and a spring-mass system. The present breakup model is based on the reasonable assumption that atomization and drop breakup are indistinguishable processes within a dense spray near the nozzle exit. Accordingly, atomization is prescribed by injecting drops which have a characteristic size equal to the nozzle exit diameter.

CHEMICAL REACTIONS

In the current version of the MAST code, it is assumed that the spray is dilute and the liquid fuel droplets act as distributed sources of fuel which evaporate to form a cloud of vapor. The combustion process thus is treated as gaseous diffusion flames. Here, an idealized approach for physically-controlled diffusion flames is to invoke a fast-chemistry assumption which the chemistry is sufficiently fast and intermediate species do not play a significant role.

In cases where chemical reactions are assumed to be in equilibrium, the equilibrium energy minimization procedures, the reader can refer to the well-known NASA Lewis CEC76 program [17]. A simplified version of equilibrium chemistry procedure based on [17] is incorporated into the MAST code for calculating mixture compositions, thermodynamics and transport properties (also refer to JANAF table [18]).

In the turbulent diffusion flame model, the influence of turbulence on combustion is taken into account by relating the fluctuations of mass fractions. This implies that fuel and oxidizer can coexist in the same place but at a different time. The effect of turbulent eddies on thermodynamical properties is modeled via the introduction of the probability function $P(\xi,x_i)$ [20]. This function contains information of both mean (f) and variance of the mixture fraction $[(f-\bar{f})^2]=g$. There variable f and g can be obtained by solving the transport equations. The density-weighted mean value ($\bar{\phi}$) of any property are evaluated by convoluting the property functions with the pdf, $P(\xi,x_i)$:

$$\bar{\phi} = \int \phi(\xi) P(\xi,x_i) d\xi$$

Numerous pdf's are available in the literature. The MAST code adopts the β-pdf which is known as the widely applicable one [21],

Chapter 3

THE COMPUTER PROGRAM

A. General Structure

The MAST family computer program consists of a set of subroutines controlled by a short main program. The fundamental structure is illustrated in Figure 3.1, showing a top to bottom block diagram encompassing the entire calculation procedure. Table 3-1 gives a brief description of the subroutines found in the MAST code. List of important FORTRAN symbols are given in Appendix A. In the following , the input file and the handling of boundary conditions are described.

The Input structure used in the MAST code utilizes blocks of optional namelists under several keywords. Each of the keywords and associated namelists are described here. These "keywords" are optional, i.e., skip if not needed.

There are ten keywords built into the current MAST code. These are : GRID, BOUND, SOLV, PROPERTY, TURBULEN, SPRAY, REACTION, RUN and ENDJOB. In addition, there is one extra block called CONTROL Block used for identifying numerics options in the code. The CONTROL block is usually created first in the input file to be read into the program through Login Unit 1. All entries are optional. If certain entries require numerical values or logical values, they are entered after the entry names separated by at least one blank space. Block name, entry names, the possible range of values and a short description of the variable are described as follow.

1. CONTROL Block : This block is always put at the top of the input file.

RESTART: It activates the usage of restart file as initial conditions by reading through LU=4.

SWIRL: It activates the swirl velocity calculations.

IMON,
JMON: It specifies the monitor point at (IMON,JMON), The default value is (2,2).

MONU,MONV, Only one variable can be specified as

MONTEMP, a monitor variable tracking on screen.
MONTK, Such as velocity at (x,y) location, pressure,
MONTE: temperature, turbulent kinetic energy,
dissipation rate. The default is MONU.

ERRCG: Convergency criterion for conjugate gradient
Matrix solver. The default value is 0.01.

NCGM: Maximum CG solver iteration number.
The default number is 100.

ERRM: Termination criterion for steady state solutions
using the time marching scheme. The default value is 0.0001.

INCOMP: Incompressible flow calculations (=1).

COMPRES: (INCOMP=0) Compressible flow options.

OMGD: 1 for supersonic flows, 0 (default) for low speed
flows.

NCRT: Number of corrector steps (default=2).

OMGF: Interpolation parameter for face velocities
1 for interpolation and 0 for averaging.
The default value is 1.

OMGT: For supersonic temperature field relaxation.

PHI: Parameter in the finite difference limiter.

OMGPHI: Parameter for Upwinding scheme.

END: End of Block input.

2. GRID Block:

NX: Number of grid points in the x direction (>3).

NY: Number of grid points in the y-direction (>3).

XLEN: It identifies the length of the calculation domain in the x direction.

YLEN: The length of the calculation domain in y direction,

READXY: It activated reading of grid point X(I,J),
and Y(I,J) from input file "sgrid.d". The input
format is given as:

```
          READ(4,9910) NX,NY
          READ(4,9920) ((X(I,J),I=1,NX),J=1,NY)
          READ(4,9920) ((Y(I,J),I=1,NX),J=1,NY)
     9910 FORMAT(2I5)
```

9920 FORMAT(10E10.4)

UNIFORM: It specified a uniform grid system.

XDIR,YDIR: It specifies an X-direction grid (only dependent on I) or a Y-direction grid (only dependent on J).Either one must be put at first of a line.

IST: The grid cell starts from IST index

IEND: The grid cell ends with IEND index

DST: The grid cell starts from DST

DEND: The grid cell ends with DEND

EXP: stretching factor

DELT: first grid cell size. EXP and DELT can be specified only once.

"power law" * Such a grid generation can be described as follows:

(a) Explicit "power law":

For example, XDIR IST 10 IEND 25 DST 3.2 DEND 5.6 EXP 1.5 The power-law formulation for such a grid distribution is,

$$X(I,J) = DST+\left(\frac{I-IST}{IEND-IST}\right)^{EXP}(DEND-DST)$$

I=IST,.........IEND for EXP>0

$$X(I,J) = DEND-\left(\frac{IEND-I}{IEND-IST}\right)^{|EXP|}(DEND-DST)$$

I=IST,.........IEND for EXP<0

. If EXP = ±1, grid is uniform

. If EXP > 1, grid is compressed close to DST and expanded close to DEND.

. If EXP < -1, grid is expanded close to DST and compressed close to DEND.

. EXP >1 and exp < -1 can be used to give a symmetric grid distribution.

(b) Implicit "power-law" :

If the user specified first grid cell size DELT instead of EXP, the grid generator automatically computes the stretching factor, EXP, and distributes the grid maintaining the first grid size DELT = X(IST +1, J) -X(IST, J) for DELT > 0 or the last grid size |DELT| =X(IEND, J)-X(IEND-1, J) for DELT < 0

3. Property block (PROP)

VISCOS: It specifies the fluid viscosity. The default value is 1.

DENGAS: It specifies the fluid density. The default value is 1.

CPGAS: It specifies the fluid specific heat. The default value is 1.

KGAS: It specifies the fluid thermal conductivity. The default value is 1.

PRG: It specifies the fluid Prandtl number. The default value is 0.74.

TEMP: It specifies the flow field initial temperature. The default value is 0.

OMEGA: Angular momentum

SMW

GAMMA $\gamma = \frac{C_P}{C_V}$

K: conductivity of the Block region

DENSITY: density

GEN: Heat generation

UIN: Initial field of U

4.Runtime block (RUN)

DT: It specifies time step.

TIME: Maximum time to be computed. DT and TIME can be specified only once.
TIME =DT *NTIME

NTIME: Maximum time step number to be computed.

NPR1: It specifies the screen monitor output frequency.

NEX: It specifies the example number. The user can code in SUBROUTINE EXAMPL.

VIN: Initial field of V

STOP: If this keyword is specified, MAST code only checks the input data and no progress is going on.

5. Variable solution block (SOLV)

U,V: Velocity field will be solved.

P Pressure will be solved.

TEMP: Temperature will be solved.

TK, TE: Turbulence kinetic energy and dissipation rate will be solved.

SW: Swirl velocity will be solved. (SWIRL must be activated in Selection block).

PATC: Particle tracking is active.

6. Turbulent block (TURBU): CT1, CT2, CMU, SME, SMK : Turbulence model constants

CT1 = 1.44,
CT2 = 1,92,
CMU =0.09,
SME =1.3,
SMK =1.0,

TKIN, TEIN : They specify the initial value of turbulence kinetic energy and dissipation rate.

SCALE: If SCALE specified, TEIN is calculated as $\mathrm{TEIN} = \mathrm{CMU} \times \mathrm{TKIN}^{1.5} / \mathrm{SCALE}$

7. Spray block (SPRAY)

SMR: It specifies the particle Sauter mean radius. If SMR < 0, it specifies a constant particle radius.

DENPT: It specifies particle density.

TEMP: It specifies particle temperature.

IST, IEND, JST, JEND: particles injected interval from (IST, JST) to (IEND, JEND)

FLOWP : It specifies spray flow rate

VINJ: It specified particle injection velocity

NPTS : It specifies particle parcels per time step

8. Boundary block (BOUND): It specifies boundary conditions in a patch format

IST, IEND, If IST = IEND or JST = JEND, it
JST, JEND : becomes a line boundary condition.

Boundary conditions specified always take such a form :
IST, IEND, JST, JEND, B.C. type, variable.

The following keywords used as B.C. type :

INLET: Inlet boundary condition, followed by variables

OUTLET: Outlet boundary condition

WALL: No split wall boundary condition, followed by variables. Wall function of turbulence model activated.

SYMMETRY : It specifies a symmetric b.c.

CYCLE : It specifies a cyclic b.c.

BLOCK : It specifies a blockage. Wall function of turbulence model activated on block face.

Variable keywords (specify b.c.0)

U : x;-direction velocity

V : y-direction velocity

TK : Turbulence kinetic energy in INLET

TE : Turbulence dissipation rate in INLET

TEMP :: Temperature in INLET WALL or BLOCK

9. Reaction block (REACTION)

NSPE : number of species to be solved.

NELE : number of elements.

NFROS : number of species to be frozen if frozen is activated.

FROSEN : activate frosen.

HP : assigned enthalpy and pressure.

TP : assigned temperature and pressure.

ELEMCONS : element conservation constants, repeat nele times.
for example: air
1 O_2 + 3.76 N_2
number of element O=1*2=2
number of element H=3.76*2=7.52
the conservation constants are 2,7.52, or 1,3.76

CHEMFORM : chemical formulation of species.

input format (4X, A8, 20F2.0), repeat NSPE times.

A8 chemical formulation

F2. 0 number of individual element presents in chemical formulation.

10. ENDJOB : ENDJOB identifies the end of the input file.

CHAPTER 4

TEST EXAMPLES

In this chapter, various sample calculations are presented. These calculations include incompressible, compressible, invicid, viscous, laminar, turbulent flows. Two-phase flows and chemical reacting flows calculations are also included.

1. Driven Cavity Flow

A square 2-D cavity with the top wall moving at a constant speed is calculated. A non-uniform grid with finer grids near the wall and a Reynolds number of 1000 are used. The input decks written on the Logic Unit (LU) 1 are listed in Figure 4.1.1. This input file consists of the Selection Block and other six blocks. Any comments can be done following a semicolon(;) at the beginning of a line or after name lists. For incompressible flow calculations, typical selections of parameters for interface interpolation and finite different schemes are given here. The U momentum was monitored at the grid point (4,3). The convergence criteria in the CG solver is 1. E-1 and the calculations are terminated at 10^{-4}.

In the GRID Block, 51 × 51 grid was used. The grid was constructed such that the first I grid point coincides with x = 0, the 26th I grid point is at x = 0.5, and the 51st I grid point is at x = 1.0. The exponential expanding scheme was used for cluster finer grid near walls with rate 1.5 as illustrated in Figure 4.1.2. The same expansion ratio and grid arrangement are the same in the y-direction. In the BOUND block, boundary conditions were identified. The top wall was represented by I grid from 1 to 51 and J = 51, U is set to be 1. East and West side boundaries were set to no slip conditions (WALL). In the PROPERTY Block, VISCOS was set to 1.0E-3 to represent Re = 1000. As can be realized, the MAST code formulates the equation from a normalized set of variables. As long as characteristic variables used to normalize the governing equations are consistent, different combinations are possible. In the SOLV block, U, V, P were identified. In the RUN block, time step (DT) was set to 1.0 and 150 time steps NTIME were marched. NPR1 was set to 10 to activate printing monitoring results, ENDJOB gave the end of input file.

Part of the program output which is written to a monitor screen for this example is included here (see Figure 4.1.3). The convergence or evolution history is printed for each time step. The entry in the first column STEP is the current time step. The second entry, TIME is the normalized time. The entry FMON display the variable being monitored at a specified point. The fourth entry indicates the variable which has the greatest change from on time step to the next. The entry FERRM is the maximum rms change of any of the variables occurring in that time step, a measure of how well the steady state being approached. If this number reduced to 10^{-3}, the solutions essentially achieve steady state.

Figure 4.1.4 shows the computed streamline contours. This case converges in one run in 83 CRAY/XMP seconds with no restarting necessary. The core memory used is 0.39 M(64-bit) words.

2. Two-D Circular Cylinder

The flow over a two-dimensional circular cylinder is an example of the external unsteady flow when Reynolds number is not too small. A 41×41 O-grid which wraps around the cylinder is used for the grid. A simple algebraic grid generation was coded in Subroutine EXAMPLE, in this case NEX 2 was used to activate the example. The grid system is shown in Figure 4.2.1. A fixed velocity was set at the outer boundary and a cyclic boundary condition was set at the front center line. The input decks are listed in Figure 4.2.2. Stream function plots at several time steps are plotted in Figure 4.2.3. More detailed calculations of this case can be refereed to [22]. 500 steps of this calculation take 175 CPU seconds and 0.29 M words core memory.

3. Laminar Backward-Facing Step Flow

The flow over a backward-facing step with a two-to-one expansion ratio was computed for a Reynolds number 800. A uniform 81 × 51 grid with blockage I from 1 to 11 and J from 1 to 26 was used, The relevant parameters used in this calculation are given in Figure 4.3.1. An experimental and theoretical study object this problem has been provided by [15]. Stream function plots of the calculation is presented in Figure 4.3.2. The calculated reattachment length at the lower wall was about 11.7 step height. This result compared favorably with other second order accurate finite difference results. The experimental data of [15] is about 14 H for this case. This case took 300 CPU seconds on a CRAY/XMP for 350 time steps and requires 0.54 MW core memory .

4. Turbulent Backward Facing Step Flow

A turbulent flow with Reynolds number 10,000 and three to two expansion ratio was calculated. A 59 × 35 non-uniform grid was used with an inlet mean velocity placed upstream of the expansion. The grid was clustered near all the solid walls using an exponentially stretching function. The standard high Reynolds number k-ε model with wall functions was used in this calculation. The relevant parameters are given in Figure 4.4.1. The grids are shown in Figure 4.4.2. The calculated stream lines are shown in Figure 4.4.3 and the calculated kinetic energy levels are shown in Figure 4.4.4. The predicted attachment length is 6.26 H. These results compared well with other results from well established codes. It took 60 CPU Seconds to run this case for 90 time steps and required 0.35 MW core memory..

5. Subsonic Channel Flow with a Circular Arc Bump.

A series of two-dimensional invicid flow calculations following the same geometries used in the literature [23,24,25] have been chosen to demonstrate the invicid flow calculations. A circular arc bump thickness-to-chord ratio of 10% is used for the subsonic case. The relevant input parameters are shown in Figure 4.5.1. For the upper and lower boundaries of the Channel, the solid-wall boundary condition for invicid calculation, i. e. zero mass flex through the surface was used. The inflow boundary is on the left hand side. For subsonic inflow conditions, total pressure and total enthalpy are held constant. For the subsonic outflow conditions, we define only the back pressure (through the initial condition PIN) and extrapolate other flow parameters from the last grid points in the calculation domain. Figure 4.5.2 and 4.5.3 present isomach lines, as well as the Mach number distributions along the top and bottom walls of the steady invicid subsonic flows solution at inlet Ma=0.5. The symmetric Mach number contours about the center chord indicates the accuracy of the numerical solution. This 65×17 solution takes 9.67 CPU seconds for 50 time steps and uses 0.24 MW sore memory on a CRAY XMP/24.

6. Transonic Flow over a Circular Bump.

This case uses the same geometry and grid points as the previous case except the inlet conditions. The relevant parameters are shown in Figure 4.6.1. For this transonic flow calculation with inlet Ma=0.675, a supersonic pocket appears in the solution and is terminated by a shocks as seen from Figure 4.6.2 and 4.6.3 in which isomach contours and surface Mach profiles are presented. The shock is captured vividly with two transition points at the bottom wall. The shock is located at a distance 73% of the chord and the maximum value of Mach value is 1.34. These agree well with the values obtained by [23,24] for the same problem. The solutions reach steady state in 120 time steps with $\Delta t=0.002$ and take 23.14 CPU seconds.

7. Supersonic Flow over a Circular Bump.

This case involves a supersonic flow with inlet Ma=1.65 past a 4% bump using a 99×33 grid. Supersonic inflow conditions requires specifications of all flow variables. For the supersonic outflow, all flow variables at the downstream boundary are extrapolated from the calculation domain points. These are done in the MAST code and are not directly reflected in the input file as shown in Figure 4.7.1. The calculated isomachs and surface Machs are shown in Figure 4.7.2 and 4.7.3 and compare well with the results of [25] using the second order Godunov method. A converged solution was obtained in 150 time steps with 51 CPU seconds and 0.46 MW core memory.

8. Supersonic Flow Past a Blunt Body.

To show the shock capturing capability involving real gases, two calculations were performed for hypersonic viscous flows, with free stream Mach number of 10.0 past a spherical nose of an axi-symmetric blunt body. The Reynolds number is 50,000 and the free stream temperature is assumed to be 300K. The standard k-ε model with wall functions was used for the calculation. The input checks for the ideal gas case is shown in Figure 4.8.1 and the calculated isomachs are shown in Figure 4.8.2. Figure 4.8.3 and 4.8.4 show the calculated temperature and Mach number along the stagnation line in front of the spherical nose.

In addition to the ideal gas case, an equilibrium chemistry package based on Gibbs free energy minimization concept has been incorporated into the present method for real gas effects. An adiabatic wall boundary condition was employed for the solutions of energy equations for both cases. In the chemical equilibrium calculation, five species including O_2, N_2, NO, N and O are involved. Concentrations of these species are calculated based on minimization of Gibbs free energy. Input decks and calculated results are shown in Figures 4.8.5, 4.8.6 and 4.8.7. This calculation was obtained by central different scheme with

50% upwind damping. To see the effect of damping, another calculation was run with 75% upwind damping (OMGPHI = 0.75). The results are shown in Figures 4.8.8 and 4.8.9. It can be seen that the shock is shaper for the 50% upwind damping case. However, the isomachs indicate oscillatory behavior for the lower, damping case. The CPU time requirements for this case are 101 seconds and 555 seconds for the ideal gas case and the equilibrium chemistry case respectively. The core memories are 0.46 MW and 0.58 MW respectively.

9. Non-Evaporating Solid-Spray

The solid-cone spray measurement of Hiroyasu and Kadato [26] have used to validate the present dense spray model. Liquid fuel is injected through a single hole nozzle into constant pressure, room-temperature nitrogen. Spray tip penetration and drop sizes were measured from photographs of the backlighted spray. The nozzle diameter was 0.3 mm and the present computations used tetradecane for the liquid fuel (the experiments used a diesel fuel oil with physical properties close to tetradecane). Atomization is introduced by injecting drops which have a characteristic size equal to the nozzle exit diameter (0.3mm).

The corresponding input data for P=1.1 MPa are shown in Figure 4.9.1. A computational domain of 20 mm in radius and 120 mm in length was discretized by a 25 radial and 45 axial grid. The mesh spacing was nonuniform with refinement on the centerline and close to the injector. The number of computational parcels was between 1000 and 3000, and the number was varying with the back pressure. The present numerical results did not change appreciably when this parcel number was varied. The initial turbulent quantities were assumed as the small values ($\kappa = 1 \times 10^{-3} m^2/s^2$, $\varepsilon = 4 \times 10^{-4} m^2/s^3$). The numerical results were insensitive to these initial values.

Figure 4.9.2 and 4.9.3 shows the spray parcel distribution, and the predicted and measured spray tip penetration versus time. It can be seen that there is reasonably good agreement between the prediction and the measurement. A detailed discussions can be found in Refs.[27, 28]. 600 steps of this calculation take 480 CPU seconds and 0.42 MW core memory.

10. Non-Evaporating Hollow-Cone Spray

The hollow-cone spray tip penetration data of Shearer and Groff [29] have been used for the model validation. In the experiment, the liquid is injected into quiescent room-temperature nitrogen at P =550 κ Pa. The corresponding input data are shown in Figure 4.10.1. The numerical timestep is used as 2.5 us and about 2000 spray parcels are used in the computation. The experimental spray cone angle is 60 degrees, and the flow rate 0.0165 mL/injection with four pulses, each of duration about 0.58 ms. The computational injection velocity are set to the experimental spray tip velocity (60m/s) measured from the movie pictures in the early stage of the injection.

Figure 4.10.2 and 4.10.3 shows the spray parcel distribution and the velocity vectors, and the predicted and measured spray tip penetration versus time. The velocity vectors show the presence of a vortex near the head of the spray, which curls the spray tip toward the outside of spray. A substantial region of strong inward flow in the center of the comer near the injector is also observed. The numerical results indicate that turbulence has a relatively small effect on penetration in a hollow-cone spray because the radial spreading of the spray dominates the spreading due to turbulence. It is seen that the predictions reasonably well agree with the experimental tip penetration. A detailed discussions can be found in Refs.[27, 28]. 528 steps of this calculation take 1020 CPU seconds and 0.67 MW core memory.

11. Hollow-Cone Spray Flames

The present numerical model for the multi-phase turbulent reacting flows has been tested by applying it to predict the local flow properties in a axisymmetric, confined, swirling spray-combusting flows. Experimental data for temperature, axial and tangential velocity components were obtained from measurement of Khalil et. al. (30). The inlet conditions and the initial droplet size distribution used in the present computations are described in Ref.[31]. The corresponding input data are shown in Figure 4.11.1. Liquid kerosene was used as fuel and the air/fuel mass ratio was fixed at 20.17.

In the present study, two swirling numbers (S = 0.72 and 1.98) were considered to investigate the influence of swirl on the droplet evaporation and burning characteristics. Figure 4.11.2, 4.11.3, and 4.11.4 show the general flow pattern such as the predicted droplet trajectories, velocity vectors, and temperature contours of two swirl cases. In the lower swirl case (S = 0.72), large portion of droplets survive in the central recirculation zone and continue to evaporate in the far downstream region. In the high swirl case (S =

1.98), most of small droplets are trapped in the recirculation zone and evaporate there, producing intensive burning and high temperature in this region. With increasing swirl, the droplet spreading increases due to the droplet dispersion and the increased particle centrifugal force term. In addition, the larger central recirculation zone corresponding to the higher inlet swirl is contributed to recirculate more hot combustion gas from downstream and to increase the temperature at near inlet regions. In this combusting dilute spray case, the present numerical procedure correctly predicts the general features of spray-combustion flows and yields the qualitative agreement with experimental data. The detailed discussions can be found from Refs.[27, 31]. The solutions reach the steady state in 1050 time steps with 2900 CPU seconds and 0.86 MW core memory.

CLOSURE

In this report, we have briefly described the methodologies and some testings of the current version of MAST code. This code was written as an advanced research-oriented computer program, in contrast to a "black-box" production code. It is advisable that the user should have some knowledge and experience with computational fluid dynamics and pressure-based methodologies. It is necessary that the user become familiar with all parts of the program before attempting to use or modify it.

Although MAST has gone through a developing and debugging phase, by no means it is a fully mature program. More advanced physical submodels, as well as new ideas of numerics will be continuously coded and tested. Documentations of further developments and improvements of MAST will be published on a timely basis as soon as their validations and testings are completed. We would also liked to ask users to contact us if any anomalies were found while working with MAST.

The research and development that went into the MAST code was supported by the NASA Mashall Space Flight Center under NASA Grants NAG8-092 and partially by NAG8-128. The authors are indebted to all of the people at NASA-MSFC that provided support, guidance, and assistance during the course of this study.

Table 1. Source Terms S^{ϕ} and $S^{\phi'}$

ϕ	S^{ϕ}	$S^{\phi'}$
u	$S^u = -\frac{1}{\rho}(g_{11}p_\xi + g_{21}p_\eta + g_{31}p_\zeta) - \frac{2}{3}(g_{11}k_\xi + g_{21}k_\eta + g_{31}k_\zeta)$ $+\frac{1}{J}(g_{11}\nu_{e\xi} + g_{21}\nu_{e\eta} + g_{31}\nu_{e\zeta})(g_{11}u_\xi + g_{21}u_\eta + g_{31}u_\zeta)$ $+\frac{1}{J}(g_{12}\nu_{e\xi} + g_{22}\nu_{e\eta} + g_{32}\nu_{e\zeta})(g_{11}v_\xi + g_{21}v_\eta + g_{31}v_\zeta)$ $+\frac{1}{J}(g_{13}\nu_{e\xi} + g_{23}\nu_{e\eta} + g_{33}\nu_{e\zeta})(g_{11}w_\xi + g_{21}w_\eta + g_{31}w_\zeta)$	$S^{u'} = [\frac{\nu_e}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})u_\eta]_\xi$ $+[\frac{\nu_e}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})u_\xi]_\eta$ $+[\frac{\nu_e}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})u_\xi]_\zeta$ $+[\frac{\nu_e}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})u_\zeta]_\xi$ $+[\frac{\nu_e}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})u_\zeta]_\eta$ $+[\frac{\nu_e}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})u_\eta]_\zeta$
v	$S^v = -\frac{1}{\rho}(g_{12}p_\xi + g_{22}p_\eta + g_{32}p_\zeta) - \frac{2}{3}(g_{12}k_\xi + g_{22}k_\eta + g_{32}k_\zeta)$ $+\frac{1}{J}(g_{11}\nu_{e\xi} + g_{21}\nu_{e\eta} + g_{31}\nu_{e\zeta})(g_{12}u_\xi + g_{22}u_\eta + g_{32}u_\zeta)$ $+\frac{1}{J}(g_{12}\nu_{e\xi} + g_{22}\nu_{e\eta} + g_{32}\nu_{e\zeta})(g_{12}v_\xi + g_{22}v_\eta + g_{32}v_\zeta)$ $+\frac{1}{J}(g_{13}\nu_{e\xi} + g_{23}\nu_{e\eta} + g_{33}\nu_{e\zeta})(g_{12}w_\xi + g_{22}w_\eta + g_{32}w_\zeta)$	$S^{v'} = [\frac{\nu_e}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})v_\eta]_\xi$ $+[\frac{\nu_e}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})v_\xi]_\eta$ $+[\frac{\nu_e}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})v_\xi]_\zeta$ $+[\frac{\nu_e}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})v_\zeta]_\xi$ $+[\frac{\nu_e}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})v_\zeta]_\eta$ $+[\frac{\nu_e}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})v_\eta]_\zeta$
w	$S^w = -\frac{1}{\rho}(g_{13}p_\xi + g_{23}p_\eta + g_{33}p_\zeta) - \frac{2}{3}(g_{13}k_\xi + g_{23}k_\eta + g_{33}k_\zeta)$ $+\frac{1}{J}(g_{11}\nu_{e\xi} + g_{21}\nu_{e\eta} + g_{31}\nu_{e\zeta})(g_{13}u_\xi + g_{23}u_\eta + g_{33}u_{e\zeta})$ $+\frac{1}{J}(g_{12}\nu_{e\xi} + g_{22}\nu_{e\eta} + g_{32}\nu_{e\zeta})(g_{13}v_\xi + g_{23}v_\eta + g_{33}v_\zeta)$ $+\frac{1}{J}(g_{13}\nu_{e\xi} + g_{23}\nu_{e\eta} + g_{33}\nu_{e\zeta})(g_{13}w_\xi + g_{23}w_\eta + g_{33}w_\zeta)$	$S^{w'} = [\frac{\nu_e}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})w_\eta]_\xi$ $+[\frac{\nu_e}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})w_\xi]_\eta$ $+[\frac{\nu_e}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})w_\xi]_\zeta$ $+[\frac{\nu_e}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})w_\zeta]_\xi$ $+[\frac{\nu_e}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})w_\zeta]_\eta$ $+[\frac{\nu_e}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})w_\eta]_\zeta$

Table 2 . Source Terms S^{ϕ} and $S^{\phi'}$ (continued)

ϕ	S^{ϕ}	$S^{\phi'}$
k	$S^k = -C_D\varepsilon + \frac{2\nu_T}{J}(g_{11}u_\xi + g_{21}u_\eta + g_{31}u_\zeta)^2$ $-\frac{2\nu_T}{J}(g_{21}v_\xi + g_{22}u_\eta + g_{32}u_\zeta)^2 + \frac{2\nu_T}{J}(g_{13}w_\xi + g_{23}w_\eta + g_{33}w_\zeta)^2$ $+\frac{\nu_T}{J}(g_{12}u_\xi + g_{22}u_\eta + g_{32}u_\zeta + g_{11}v_\xi + g_{21}v_\eta + g_{31}v_\zeta)^2$ $+\frac{\nu_T}{J}(g_{13}u_\xi + g_{23}u_\eta + g_{33}u_\zeta + g_{11}w_\xi + g_{21}w_\eta + g_{31}w_\zeta)^2$ $+\frac{\nu_T}{J}(g_{13}v_\xi + g_{23}v_\eta + g_{33}v_\zeta + g_{12}w_\xi + g_{22}w_\eta + g_{32}w_\zeta)^2$	$S^{k'} = [\frac{\nu_{ek}}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})k_\eta]_\xi$ $+[\frac{\nu_{ek}}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})k_\xi]_\eta$ $+[\frac{\nu_{ek}}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})k_\xi]_\zeta$ $+[\frac{\nu_{ek}}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})k_\zeta]_\xi$ $+[\frac{\nu_{ek}}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})k_\zeta]_\eta$ $[\frac{\nu_{ek}}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})k_\eta]_\zeta$
ε	$S^\varepsilon = \frac{\varepsilon}{k}(C_1 Pr - C_2\varepsilon)$	$S^{\varepsilon'} = [\frac{\nu_{e\varepsilon}}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})\varepsilon_\eta]_\xi$ $+[\frac{\nu_{e\varepsilon}}{J}(g_{11}g_{21} + g_{12}g_{22} + g_{13}g_{23})\varepsilon_\xi]_\eta$ $+[\frac{\nu_{e\varepsilon}}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})\varepsilon_\xi]_\zeta$ $+[\frac{\nu_{e\varepsilon}}{J}(g_{11}g_{31} + g_{12}g_{32} + g_{13}g_{33})\varepsilon_\zeta]_\xi$ $+[\frac{\nu_{e\varepsilon}}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})\varepsilon_\zeta]_\eta$ $+[\frac{\nu_{e\varepsilon}}{J}(g_{21}g_{31} + g_{22}g_{32} + g_{23}g_{33})\varepsilon_\eta]_\zeta$

TABLE 3-1, Descriptions of Subroutine

MAIN

Program MAIN handles the main control and sequence of prediction-corrector in each time step. It first calls routines INITIL and TRANSF for setting up initial values and coefficients of grid transformation and then enters the main time-marching loop. Inside this loop, the predictor-corrector sequence is carried out. After the marching process is finished, MAIN also handles the writing of the solution to disks by calling OUTPUT.

AUXSUB (NCONTL)

This is an auxiliary subroutine to carry out further some link coefficients for pressure increment equations (25) and (29). NCONTL = 1 and 2 are called from momentum equation to calculate phase values of H() in equation (24) and (27). NCONTL =3 and 4 are used for calculations of boundary pressure values after 1st corrector and second corrector step. NCONTL =5 is used again for boundary pressures after all predictor- corrector steps are done.

CADPXY (NCONTL)

This is a utility subroutine to calculate pressure gradients for non-orthogonal grid.

CALCAP

This subroutine computes link coefficients for all interior pressure-increment quantities except boundary points which are done by calling UPDTCF(2).

CALCTB (NCONTL)

This subroutiine computes corrector step of turbulent quantities k and ε using explicit schemes. Boundary points are calculated through calling UPDTBC (4,4) and UPDTBC (4,5).

CALCVU (NCONTL)

This subroutine calculates both velocity components through 1st corrector step of (24) and 2nd corrector step of (27) using explicit expressions. NCONTL = 1 represents 1st corrector and NCONTL = 2 represents 2nd corrector step.

CGSOL V(NCONTL),
MATRIX, and DOTSUM

These subroutines activate the Conjugate gradient solver solution procedure for algebraic equations derived from the discretization procedure.NCONTL = 0 is used for pressure increment finite difference equation solver and NCONTL = 1 is for the rest of the variables.

COEFIS

This subroutine calculates link coefficients A_p and A_{nb} for all transport equations which have the common form $A_pf_p= SA_{nb}f_{nb}+ S_u$ except two pressure incremental equations (25) and (29). Depending on the finite difference form used for discretization of convection terms, the link coefficients are constructed differently. It also calls UPDTCF for link coefficients of boundary points and MODCOF as well as MODSOR if modifications of these coefficients and source terms,which have the form S_pf_p , are required.

EXAMPL

Several tested example cases are supplied in this subroutine for the user to familiarize the current code.

GEOMBC

This subroutine calculates direction cosines and normal distance of the first interior points to the solid boundaries.

INITIL

This subroutine initializes most of the variables and arrays and reads in the input namelist to obtain the user-specified parameters. It calls KEYREAD to read in the key parameters (which will be described in KEYREAD). It contains some utilities such as reading in the restart file, grid systems if grids were generated by the user through other generation codes. Any blockage of calculation domain will also be done in this subroutine thus it calls GEOMBC to modify interior points.

MODIFY

This subroutine gives user the freedom to create or modify boundary conditions, source terms and coefficients that are not included in the current version of the MAST code.

NEWSTP

The NEWSTP subroutine puts the most current calculated variables (i.e. variables at t^{n+1}) to the old time step before continuing the next time step cycle.

SORCEM (NCONTL),

These subroutines set up source terms for the

SORCEP (NCONTL), SORCSW, SORCTH, SORCET (NCONTL)

momentum eequation (NCONTL = 1 for u, 2 for v), pressre correction (NCONTL =1 for 1st corrector, 2 for second corrector), swirl velocity, thermal energy equation and turbulence quantities (NCONTL = 1 for k and 2 for ε) respectively.

SPRAY

This subroutine calculates spray droplets dynamics based on a lagrangian tracking scheme.

SYSTEM

Subroutine SYSTEM is the main control routine of the discretization method used in the MAST code. It composes of various ENTRY's for setting up coefficients and sources of finite difference schemes, for solving the nonlinear algebraic equations and updating boundary conditions. For example, entry PREDIT solves equation (22) for velocity components in x direction (NF = 1) and y direction (NF = 2). It calls COEFIS to set up pressure terms, cross derivative diffusion terms and extra terms (such as swirl source terms and rotations if desired).

The calling to CGSOLV is to utilize the conjugate gradient method for solveing the resulting discretized algebraic equations.

Entry FSTCRT solves the incremental pressure (equation (25)) and incremental velocity equation (equation(24)). Note that equation (24) is explicit after pressure increment being solved thus by simply calling CALCVU is sufficient. Entry OTHCRT is deviced for second corrector step or other corrector steps if high order time accurate discretization in temporal domain were used. Entries THERML and SWIRL are used for thermal energy equation in heat transfer prolems and Swirling Velocity Component (w). Entries PREDTB and FCRTTB are predictor and corrector procedures for turbulence quantities such as kinetic energy equation (k) and its dissipation rate (ε) equation if the $k-\varepsilon$ two equation model were used for turbulent flow calculation. It should be noted that the predictor step for turbrlent quantities is implicit and the corrector step is explicit.

TRANSF

This subroutine carries out grid transformation coefficients, and Jacobians by a central difference scheme.

UPDTBC
(NCONTL, NTUB)

This subroutine updates various boundary conditions for fluid flow and heat transfer problems. When NCONTL is set to be 1, it updates outflow boundary conditions based on mass conservation balance.
NCONTL = 2 represents symmetric ($\frac{\partial \phi}{\partial X_i} = 0$) or slip boundary conditions. NCONTL = 3 sets up cyclic boundary conditions. If wall function options were used in the turbulent flow calculation, NCONTL should be set to 4. Under this option, NTUB = 1 recalculates an effective viscosity at the first point away from the wall. NTUB = 2 and 3 re-establish new source terms for k and ε equation in the first corrector step. NTUB = 4 and 5 create source terms for explicit $k-\varepsilon$ equation corrector step. NCONTL = 5 is used for adabatic boundary condition in thermal energy equation.

UPDTCF (NCONTL)

This subroutine is used for updating the link coefficients for boundary points specifically for body-fitted coordinates when the grid lines do not intersect boundary orthogonally. NCONTL = 1 activates this action for all variables except pressure. NCONTL = 2 is used for pressure.

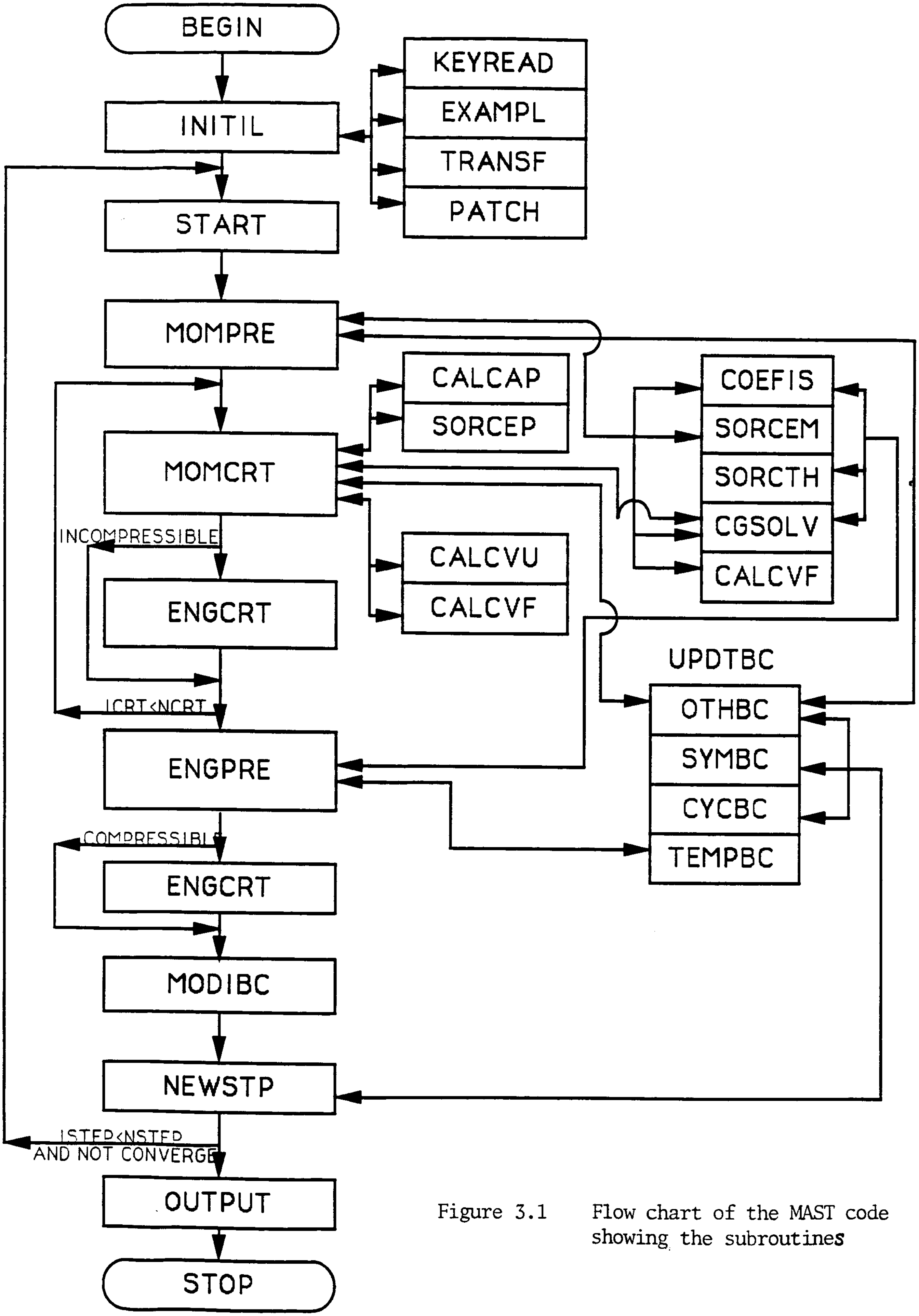


Figure 3.1 Flow chart of the MAST code showing the subroutines

```
CONTROL  NCRT 2
INCOMP OMGD 0 OMGF 0.4 PHI -1.000 OMGPHI 0.00
IMON 4 JMON 3 MONU  ERRCG 1.E-1  ERRM 1.E-4
;RESTART
GRID   NX   51 NY 51
     XDIR  IST  1  IEND  26  DST  0.0  DEND  0.5  EXP  1.50
     XDIR  IST 26  IEND  51  DST  0.5  DEND  1.0  EXP -1.50
     YDIR  IST  1  IEND  26  DST  0.0  DEND  0.5  EXP  1.50
     YDIR  IST 26  IEND  51  DST  0.5  DEND  1.0  EXP -1.50
BOUND
   IST   1 IEND   51 JST   51 JEND  51  WALL  U 1.
   IST   1 IEND   51 JST    1 JEND   1  WALL
   IST  51 IEND   51 JST    1 JEND  51  WALL
   IST   1 IEND    1 JST    1 JEND  51  WALL
PROPERTY  VISCOS  1.0E-3
SOLV      U    V    P
RUN       DT 1.0    NSTEP 150  NPR1 10
ENDJOB
```

Figure 4.1.1 Input deck for the driven cavity flow

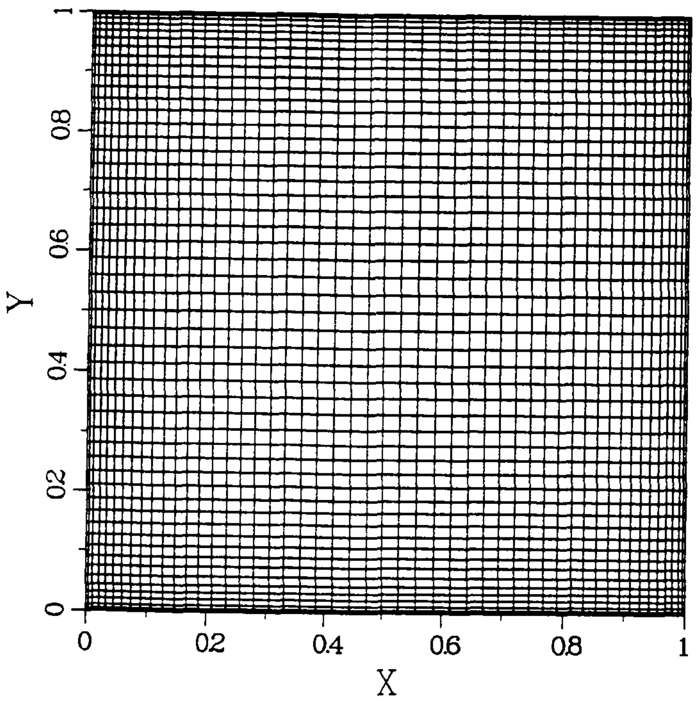


Figure 4.1.2 Grids for the driven cavity flow

```
          CONVERGENCE HISTORY FOR DRIVEN CAVITY FLOW (Re=1000)

STEP=    10 TIME=1.000E+01 FMON=-4.142E-05 NF=    2 FERRM=1.023E-01
STEP=    20 TIME=2.000E+01 FMON= 6.257E-05 NF=    2 FERRM=3.528E-02
STEP=    30 TIME=3.000E+01 FMON= 1.794E-04 NF=    2 FERRM=1.546E-02
STEP=    40 TIME=4.000E+01 FMON= 2.568E-04 NF=    2 FERRM=5.971E-03
STEP=    50 TIME=5.000E+01 FMON= 2.766E-04 NF=    1 FERRM=2.384E-03
STEP=    60 TIME=6.000E+01 FMON= 2.833E-04 NF=    2 FERRM=1.424E-03
STEP=    70 TIME=7.000E+01 FMON= 2.878E-04 NF=    2 FERRM=9.368E-04
STEP=    80 TIME=8.000E+01 FMON= 2.896E-04 NF=    2 FERRM=6.548E-04
STEP=    90 TIME=9.000E+01 FMON= 2.888E-04 NF=    2 FERRM=4.813E-04
STEP=   100 TIME=1.000E+02 FMON= 2.893E-04 NF=    2 FERRM=3.512E-04
STEP=   110 TIME=1.100E+02 FMON= 2.895E-04 NF=    2 FERRM=2.619E-04
STEP=   120 TIME=1.200E+02 FMON= 2.898E-04 NF=    2 FERRM=1.998E-04
STEP=   130 TIME=1.300E+02 FMON= 2.898E-04 NF=    2 FERRM=1.490E-04
STEP=   140 TIME=1.400E+02 FMON= 2.901E-04 NF=    2 FERRM=1.120E-04
```

Figure 4.1.3 Sample monitors for the driven cavity flow

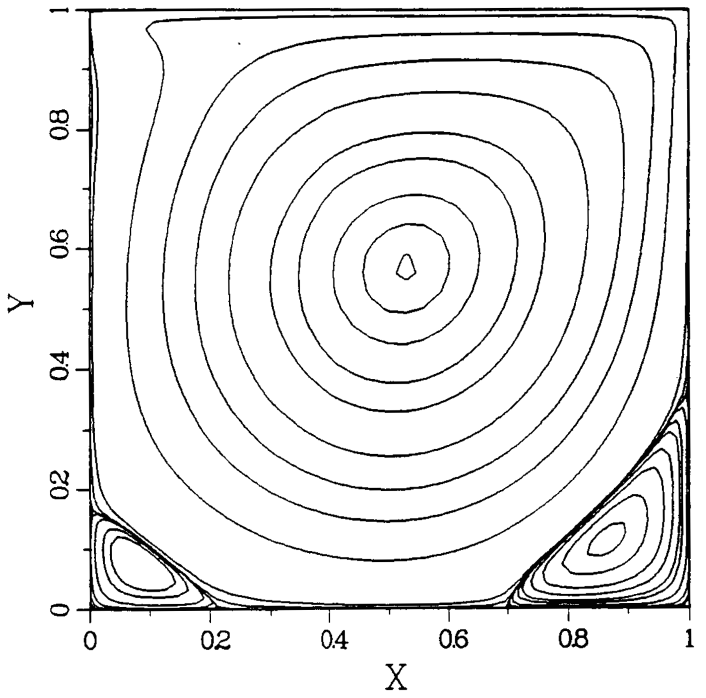


Figure 4.1.4 Computed streamlines for the driven cavity flow

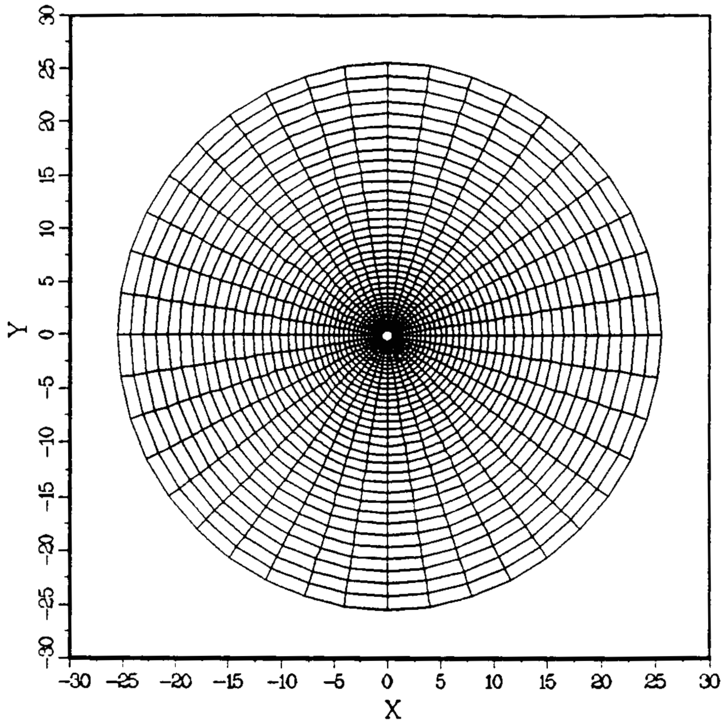
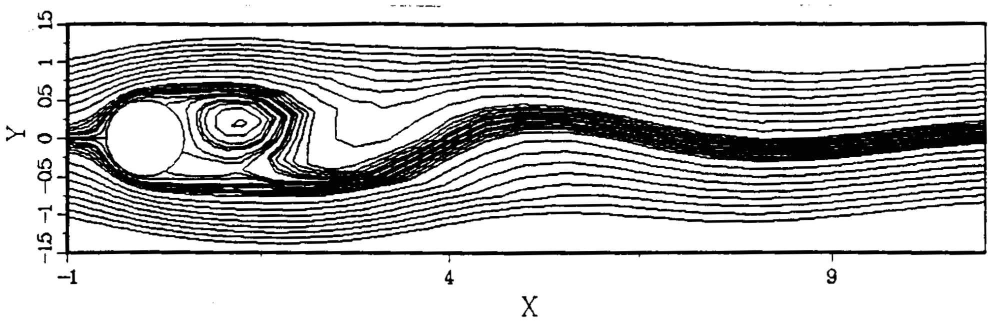


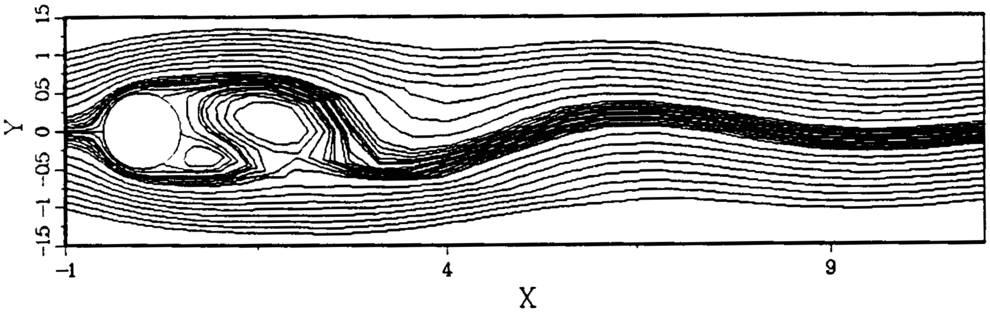
Figure 4.2.1 Grid system for a 2-D flow over a circular cylinder

```
CONTROL  NCRT 2
INCOMP OMGD 0.   PHI -1.0  OMGPHI 0.00  OMGF 1.0
IMON 4   JMON 3   MONU  ERRCG 1.E-2 ERRM 1.E-8
;RESTART
GRID  NX 41  NY 41
;GRID SET IN SUBROUTINE EXAMPLE
BOUND
     IST  1  IEND   41 JST  1 JEND   1  WALL
     IST  1  IEND   41 JST 41 JEND  41  WALL  U 1.
     IST  1  IEND   1  JST  1 JEND  41  CYCLE
PROPERTY  VISCOS  5.0E-3   ;RENOLDS NUMBER=200
SOLV      U   V    P
RUN       DT 0.19      NSTEP 500 NPR1 10   NEX 2
     ;    DT 0.02625   NSTEP 50  NPR1 10   NEX 2
ENDJOB
```

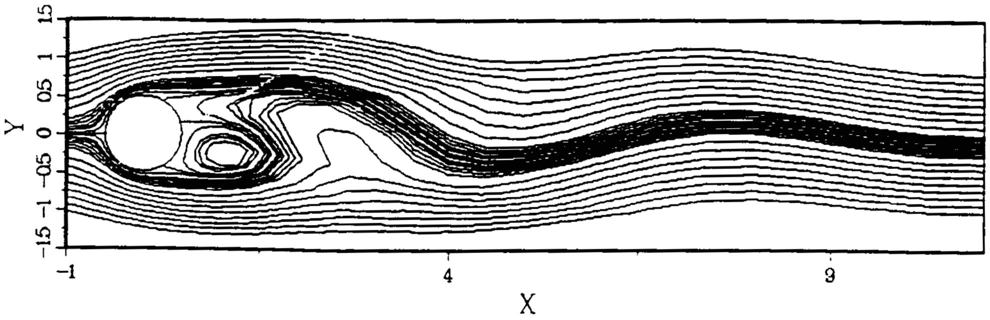
Figure 4.2.2 Input deck for the cylinder flow



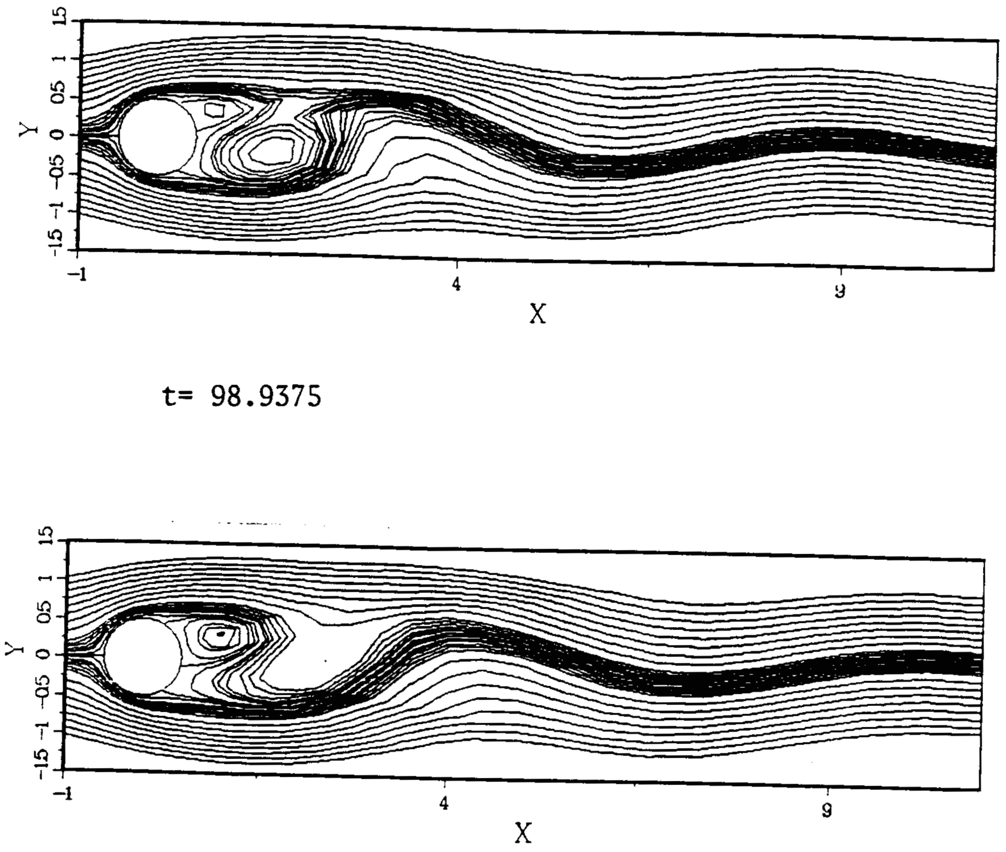
t= 95 sec



t= 96.3125



t= 97.625



t= 98.9375

t= 100.25

Figure 4.2.3 Streamline Patterns at different times within a shadding cycle

```
;------------------- BACKWARD FACING STEP,LAMINAR FLOW -----------------
CONTROL
INCOMP OMGD 0 PHI -1.00 OMGPHI 0.0 ERRCG 1.E-1 ERRM 1.E-4
IMON 17 JMON 4 MONU  OMGF 0.2
 NCRT 3
;RESTART
GRID   NX  81 NY 51
     XDIR  IST  1  IEND  81  DST  0.0  DEND  24.  EXP +1.00
     YDIR  IST  1  IEND  51  DST  0.0  DEND  1.0  EXP +1.00
BOUND
   IST   1 IEND  11  JST   1  JEND  26  BLOCK
   IST   1 IEND   1  JST  27  JEND  51  INLET   U 1
   IST  81 IEND  81  JST   1  JEND  51  OUTLET
   IST  12 IEND  81  JST   1  JEND   1  WALL
   IST   1 IEND  81  JST  51  JEND  51  WALL
PROPERTY  VISCOS 1.25E-3 DENGAS 1. ;RENOLDS NUMBER=800
SOLV      U    V    P
RUN       DT 1.25     NSTEP 350 NPR1 10 NEX 1
ENDJOB
```

Figure 4.3.1 Input deck for the Laminar Backward Step Flow

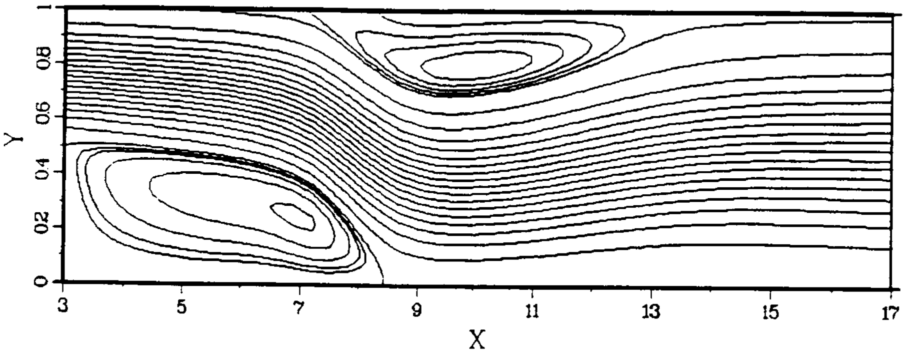


Figure 4.3.2 Contour of Streamlines for the Laminar Backward Facing Step Flow

```
CONTROL NCRT 3
INCOMP OMGD 0 PHI -1.0 OMGPHI .30 ERRCG 1.E-1 ERRM 1.E-4
;RESTART
IMON   15  JMON 4     MONU
GRID   NX 59   NY 35   ;XLEN  18. YLEN 1.
       XDIR IST  1  IEND 11   DST 0.      DEND  1.5         EXP -1.5
       XDIR IST 11  IEND 40   DST 1.5     DEND  4.5         EXP  1.2
       XDIR IST 40  IEND 59   DST 4.5     DEND  18.         EXP  1.5
       YDIR IST  1  IEND 15   DST 0.      DEND  0.3333      EXP  1.0
       YDIR IST 15  IEND 25   DST 0.3333 DEND  0.6666      EXP  1.5
       YDIR IST 25  IEND 35   DST 0.6666 DEND  1.0         EXP -1.5
BOUND
        IST    1 IEND  59 JST  1 JEND  1  WALL
        IST    1 IEND  59 JST 35 JEND 35  WALL
        IST    1 IEND   1 JST 16 JEND 35  INLET U 1
        IST   59 IEND  59 JST  1 JEND 35  OUTLET
        IST    1 IEND  11 JST  1 JEND 15  BLOCK
PROPERTY  VISCOS  1.000E-5   ;RENOLDS NUMBER=1.E5
TURBULEN  TKIN 0.003  SCALE 0.33
SOLV     U    V    P    TK    TE ;TURBULENCE MODEL ACTIVATED
RUN       DT 0.6 NSTEP 90 NPR1 10
ENDJOB
```

Figure 4.4.1   Input deck for the turbulent backward-facing step flow

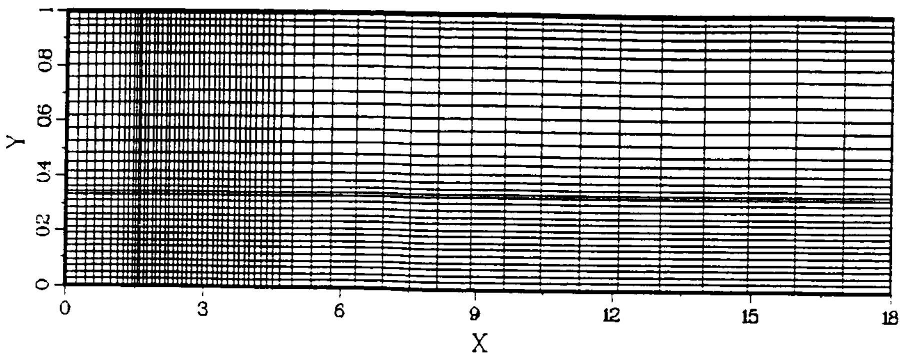


Figure 4.4.2 Grid system for the backward-facing step flow

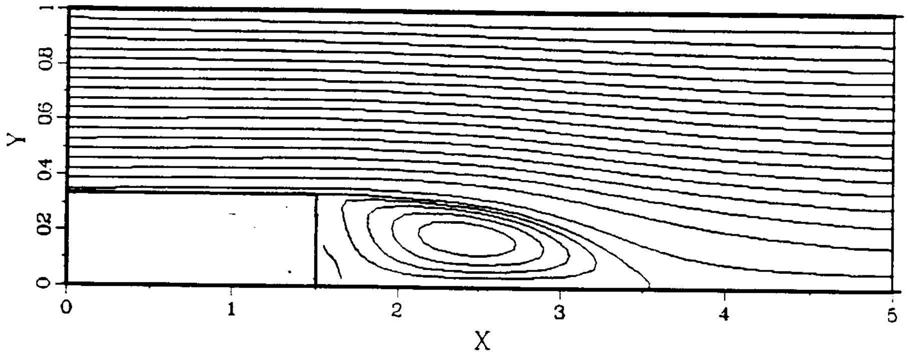


Figure 4.4.3 Streamline contours for the turbulent backward-facing step flow

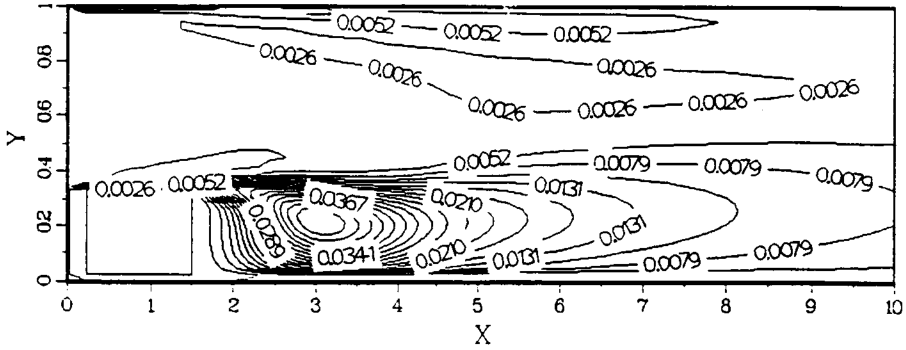


Figure 4.4.4 Turbulent kinetic energy contours of the backward-facing step flow

```
CONTROL
 COMPRES OMGD 0.0 PHI +0.33 OMGPHI 0.00  OMGT 0.5 OMGF 0.1
 NCRT 3  ERRCG 1.E-1  ERRM 1.E-4 IMON 4 JMON 3 MONU
;RESTART
GRID   NX  65 NY 17
BOUND
   IST  1  IEND   1  JST   1  JEND  17  INLET
   U 169.37754   TEMP 285.7143  P   1.0E5   ;Min=0.5
   IST  65 IEND   65 JST    1 JEND  17  OUTLET
   IST  1  IEND   65 JST    1 JEND   1  SLIP  Q 0.
   IST  1  IEND   65 JST   17 JEND  17  SYMMETRY Q 0.
PROPERTY  VISCOS 0.E-3
   UIN 169.37754 TIN 285.7143 PSTAG 1.18621E5 TSTAG 300 PIN 1.0E5
SOLV       U    V    P   TEMP
RUN        DT 3.0E-3   NSTEP 50 NPR1 10  NPR2 600   NEX 8
ENDJOB
```

Figure 4.5.1 Input deck for the subsonic bump flows

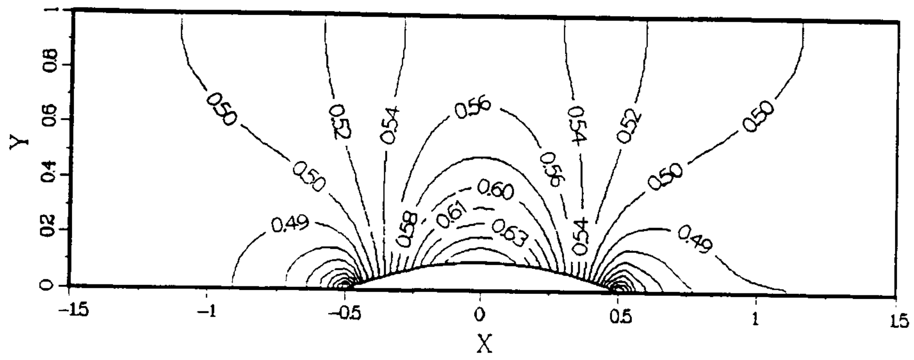


Figure 4.5.2 IsoMach line contours of the subsonic bump flow

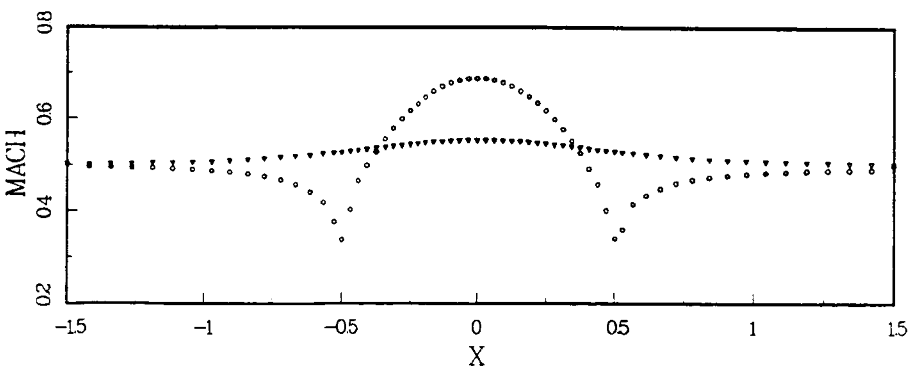


Figure 4.5.3 Mach number distributions along the walls

```
CONTROL
 COMPRES OMGD 1.0 PHI +0.90 OMGPHI 0.15  OMGT 0.8 OMGF 0.10
 NCRT 3  ERRCG 1.E-1  ERRM 1.E-4 IMON 4 JMON 3 MONU
;RESTART
GRID   NX  65 NY 17
BOUND
   IST  1  IEND   1  JST   1  JEND  17  INLET
   U 221.20149   TEMP 274.9456  P   1.0E5   ;Min=0.675
   IST  65 IEND   65 JST    1 JEND  17  OUTLET
   IST  1  IEND   65 JST    1 JEND   1  SLIP  Q 0.
   IST  1  IEND   65 JST   17 JEND  17  SYMMETRY Q 0.
PROPERTY  VISCOS 0.E-3
   UIN 221.20149 TIN 274.9456 PSTAG 1.35694E5 TSTAG 300 PIN 1.0E5
SOLV       U    V    P   TEMP
RUN        DT 2.0E-3   NSTEP 120 NPR1 10  NPR2 600   NEX 8
ENDJOB
```

Figure 4.6.1 Input Deck for the Transonic Bump Flow

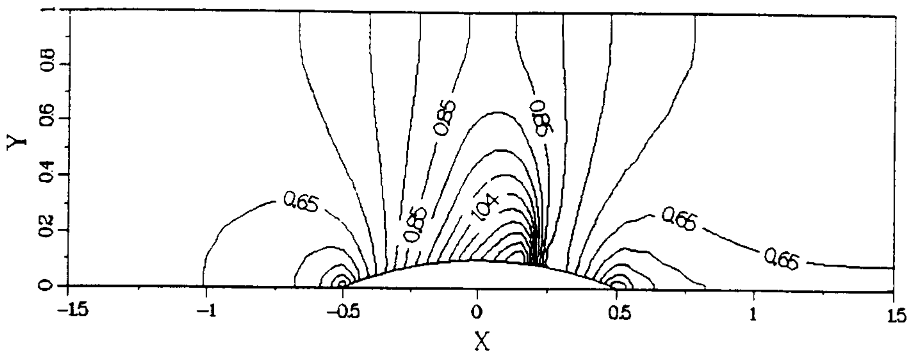


Figure 4.6.2 IsoMach Contours for the Transonic Bump Flow

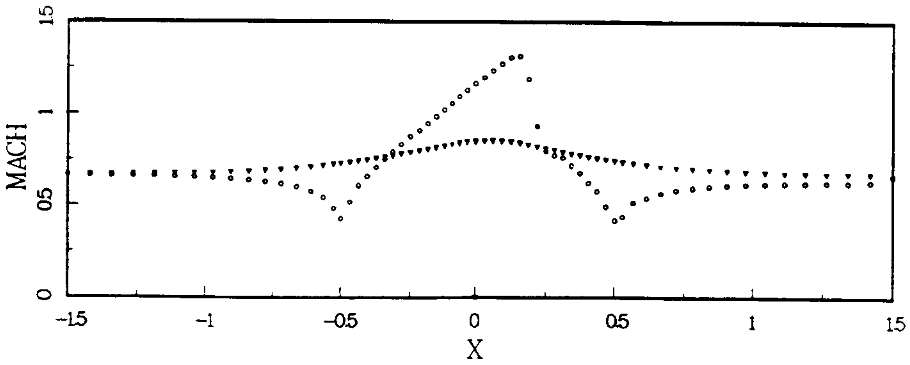


Figure 4.6.3 Surface Mach Numbers Along the Walls

```
CONTROL
 COMPRES OMGD 1.0 PHI +0.33 OMGPHI 0.10  OMGT 0.5 OMGF 0.5
 NCRT 3  ERRCG 1.E-1  ERRM 1.E-4 IMON 4 JMON 3 MONU
;RESTART
GRID   NX  99 NY 33
BOUND
   IST  1  IEND   1  JST   1  JEND  33  INLET
   U 460.86153   TEMP 194.2376  P    1.0E5   ;Min=1.65 DT 0.3E-3 OMGT 0.5
   IST  99 IEND   99 JST    1 JEND  33  OUTLET
   IST  1  IEND   99 JST    1 JEND   1  SLIP  Q 0.
   IST  1  IEND   99 JST   33 JEND  33  SYMMETRY Q 0.
PROPERTY  VISCOS 0.E-3
   UIN 460.86153 TIN 194.2376 PSTAG 4.57886E5 TSTAG 300 PIN 1.0E5
SOLV      U     V     P    TEMP
RUN       DT 0.3E-3   NSTEP 150 NPR1 10  NPR2 600   NEX 9
ENDJOB
```

Figure 4.7.1 Input Deck for the Supersonic Bump Flow

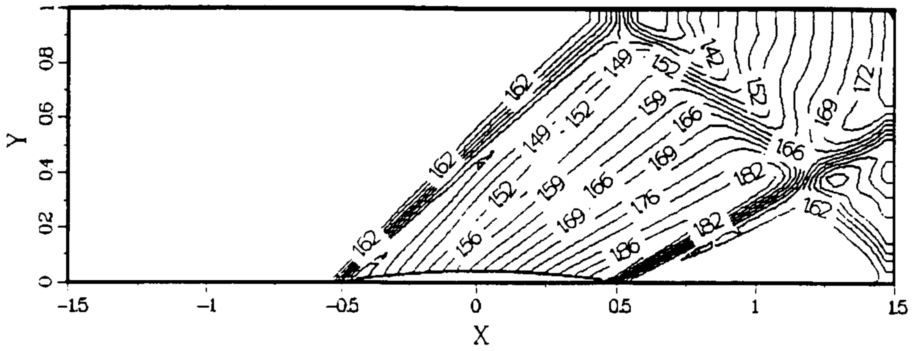


Figure 4.7.2 IsoMach Contours of the Supersonic Pump Flow

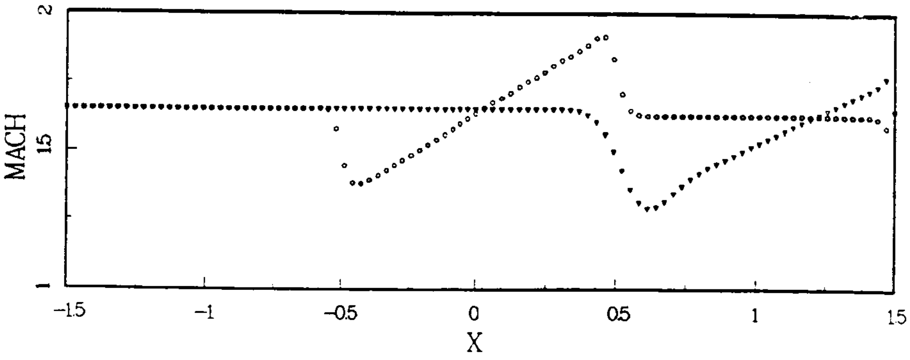


Figure 4.7.3 Surface Mach Numbers Along the Walls

```
CONTROL   NCRT 3
 COMPRES OMGD 1 PHI +1.0 OMGPHI 0.50 OMGT 0.3  OMGF .00
;FOR EQUILIBRIUM OMGPHI=0.75
 IMON 4 JMON 3 MONU  ERRCG 1.E-1  ERRM  1.E-4
;RESTART
GRID   NX  81 NY 41    AXISYM
BOUND
   IST  1  IEND   1  JST    1 JEND  41  SYMMETRY Q 0
   IST 81  IEND  81  JST    1 JEND  41  OUTLET
   IST  1  IEND  81  JST    1 JEND   1  WALL U 0. V 0.
;  IST  1  IEND  81  JST    1 JEND   1  SLIP   Q 0
   IST  1  IEND  81  JST   41 JEND  41  INLET
   U 3471.269240 TEMP 300. P 1.658E4 TK 10000. TE 30000000.
PROPERTY  VISCOS 2.00E-5
   UIN 3471.269240 TIN 300. PSTAG 7.1510109E8 TSTAG 6300. PIN 1.658E4
TURBULEN
   TKIN 10000. TEIN 30000000. ; SCALE 1.0
SOLV       U    V    P  TEMP   TK  TE
RUN        DT 1.5E-4  NSTEP  50   NPR1 10   NEX 13
ENDJOB
```

Figure 4.8.1 Input Deck for the Hypersonic Flow Past a Blunt Body-Ideal Gas

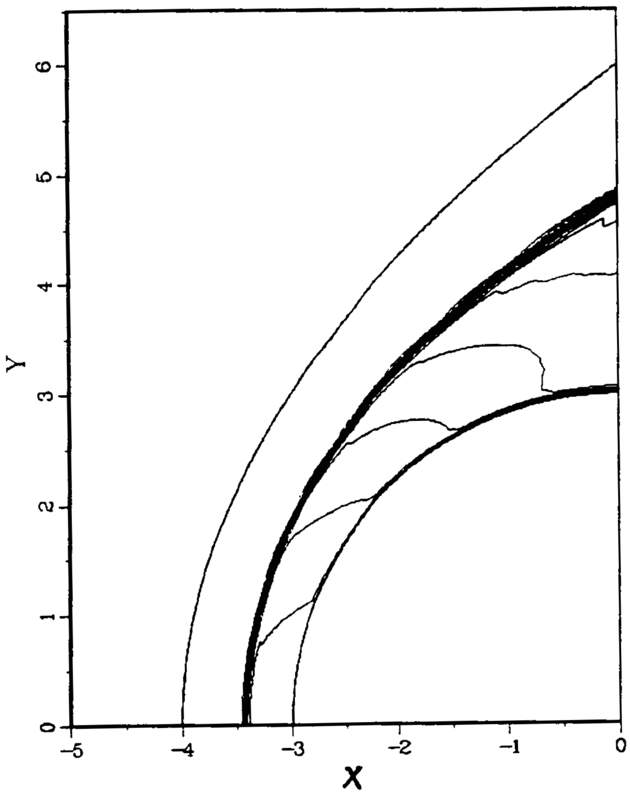


Figure 4.8.2 Mach Number Contours for the Blumt Body Flow

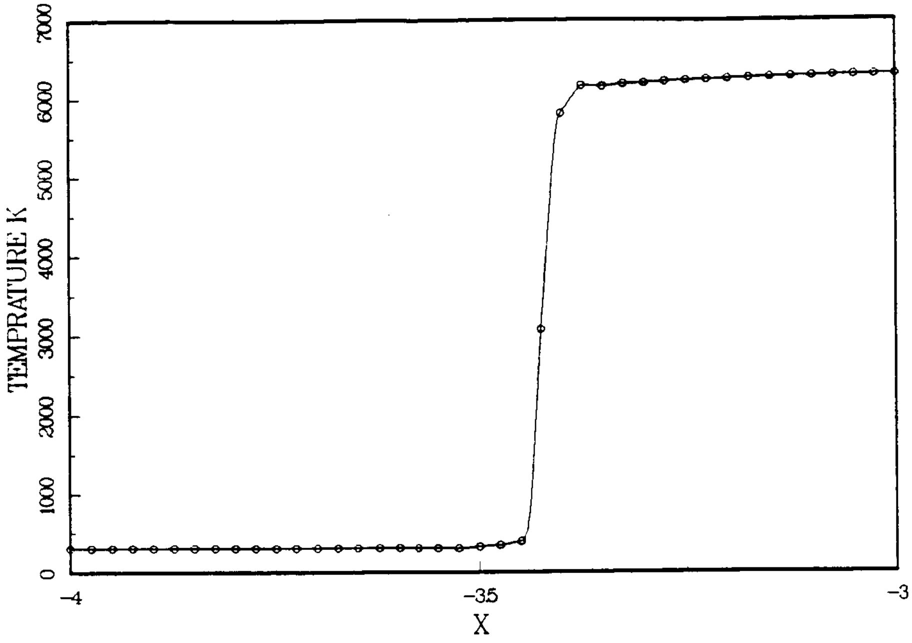


Figure 4.8.3 Temperature Profile Along the Stagnation Line

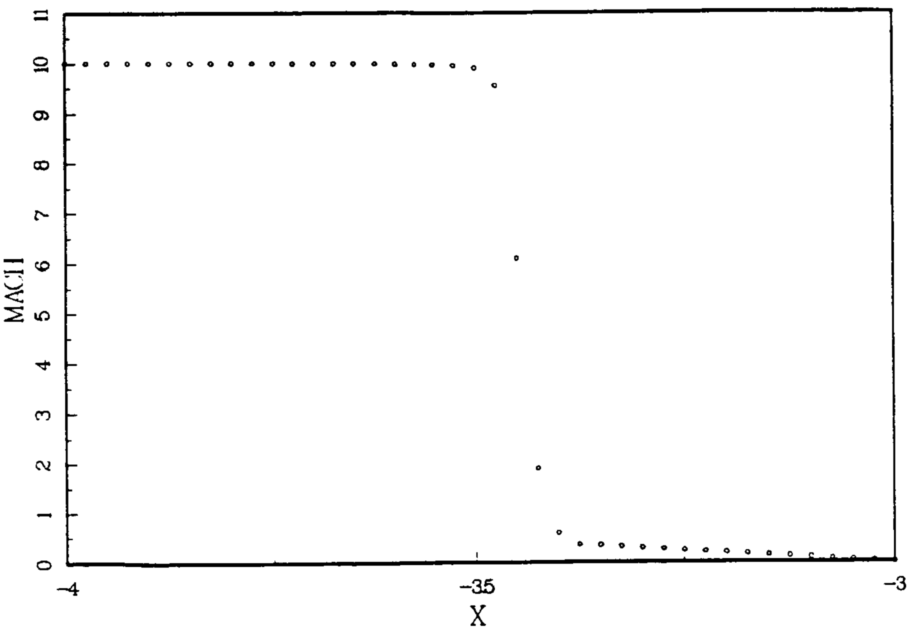


Figure 4.8.4 Mach Number Along the Stagnation Line

```
CONTROL   NCRT 3
 COMPRES OMGD 1 PHI +1.0 OMGPHI 0.50 OMGT 0.3  OMGF .00
;FOR EQUILIBRIUM OMGPHI=0.75
 IMON 4 JMON 3 MONU  ERRCG 1.E-1  ERRM  1.E-4
;RESTART
GRID   NX  81 NY 41   AXISYM
BOUND
   IST  1  IEND   1  JST    1 JEND  41  SYMMETRY Q 0
   IST  81 IEND  81  JST    1 JEND  41  OUTLET
   IST  1  IEND  81  JST    1 JEND   1  WALL U 0. V 0.
;  IST  1  IEND  81  JST    1 JEND   1  SLIP   Q 0
   IST  1  IEND  81  JST   41 JEND  41  INLET
   U 3471.269240 TEMP 300. P 1.658E4 TK 10000. TE 30000000.
PROPERTY  VISCOS 2.00E-5
   UIN 3471.269240 TIN 300. PSTAG 7.1510109E8 TSTAG 6300. PIN 1.658E4
TURBULEN
   TKIN 10000. TEIN 30000000. ; SCALE 1.0
REACTION
    NSPE 5  NELE 2  NFROS 2  HP ;FROSEN  TP  HP
    ELEMCONS  1  3.76
    CHEMFORM  ;INPUT DATA FORMAT (4X,A8,8F2.0)
    O2       2 0
    N2       0 2
    NO       1 1
    N        0 1
    O        1 0
SOLV   CONHT  EQLM   U   V   P  TEMP   TK  TE
RUN       DT 1.5E-4  NSTEP  150   NPR1 10  NEX 13
ENDJOB
```

Figure 4.8.5   Input deck for hypersonic flow-equilibrium air

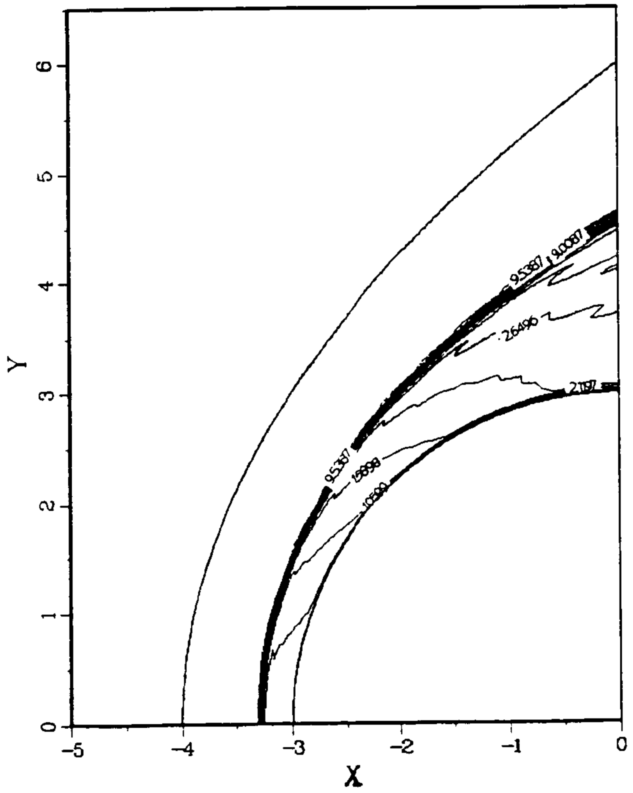


Figure 4.8.6   Contour of Mach number
50% damping

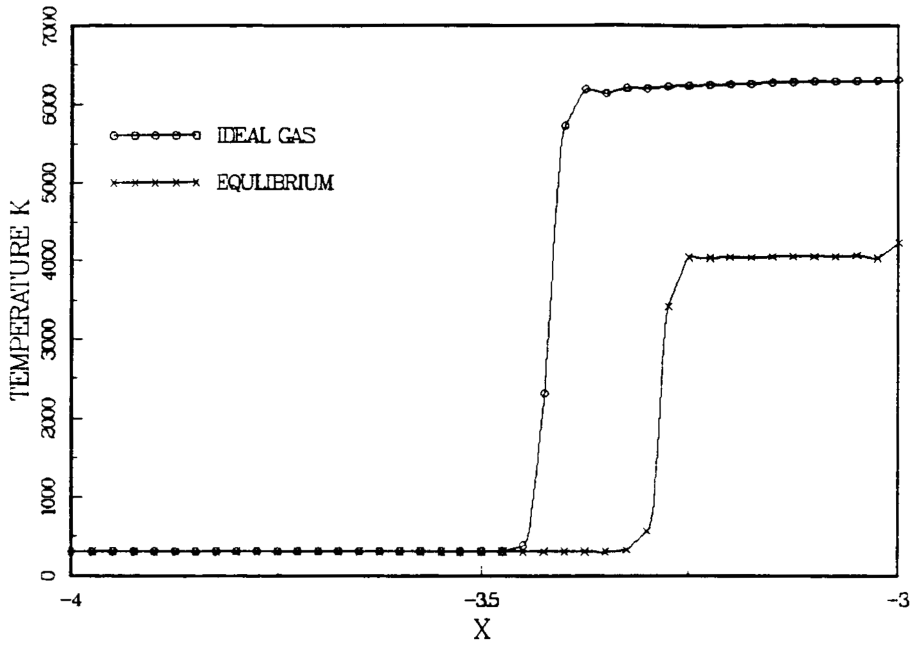


Figure 4.8.7 Stagnation Line Temperature for Ideal Gas Air and Equilibrium air, 50% Damping

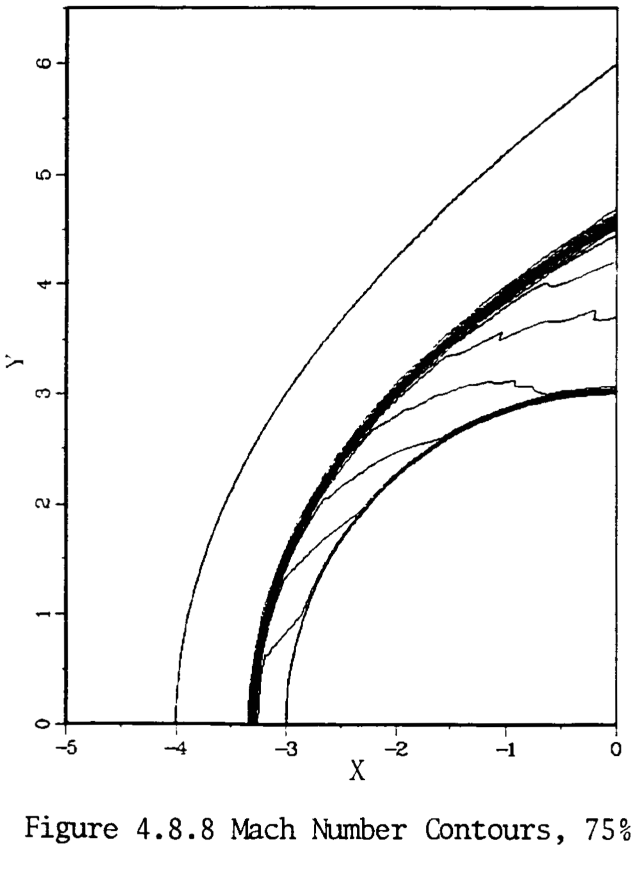


Figure 4.8.8 Mach Number Contours, 75% Damping

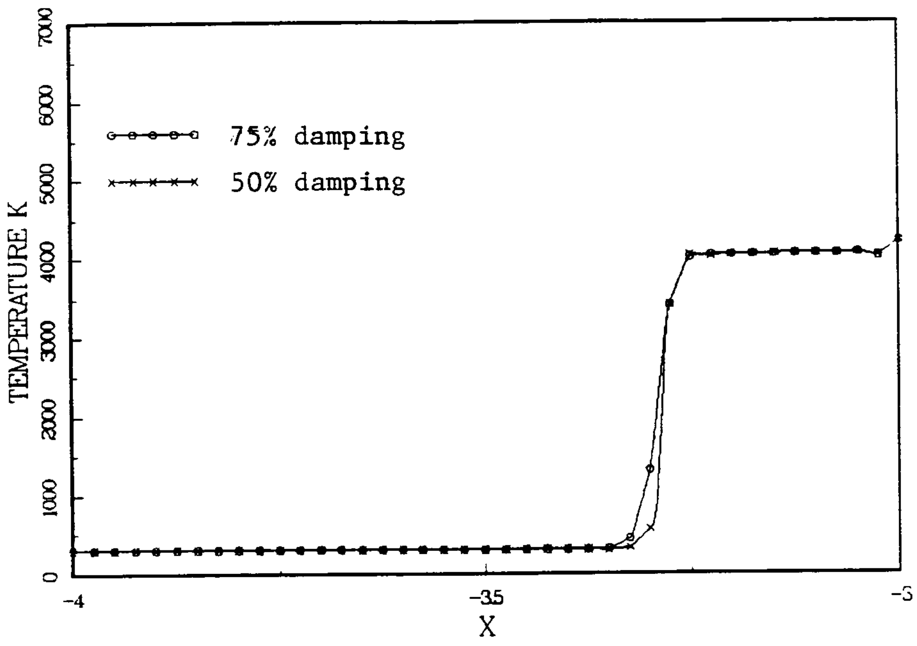


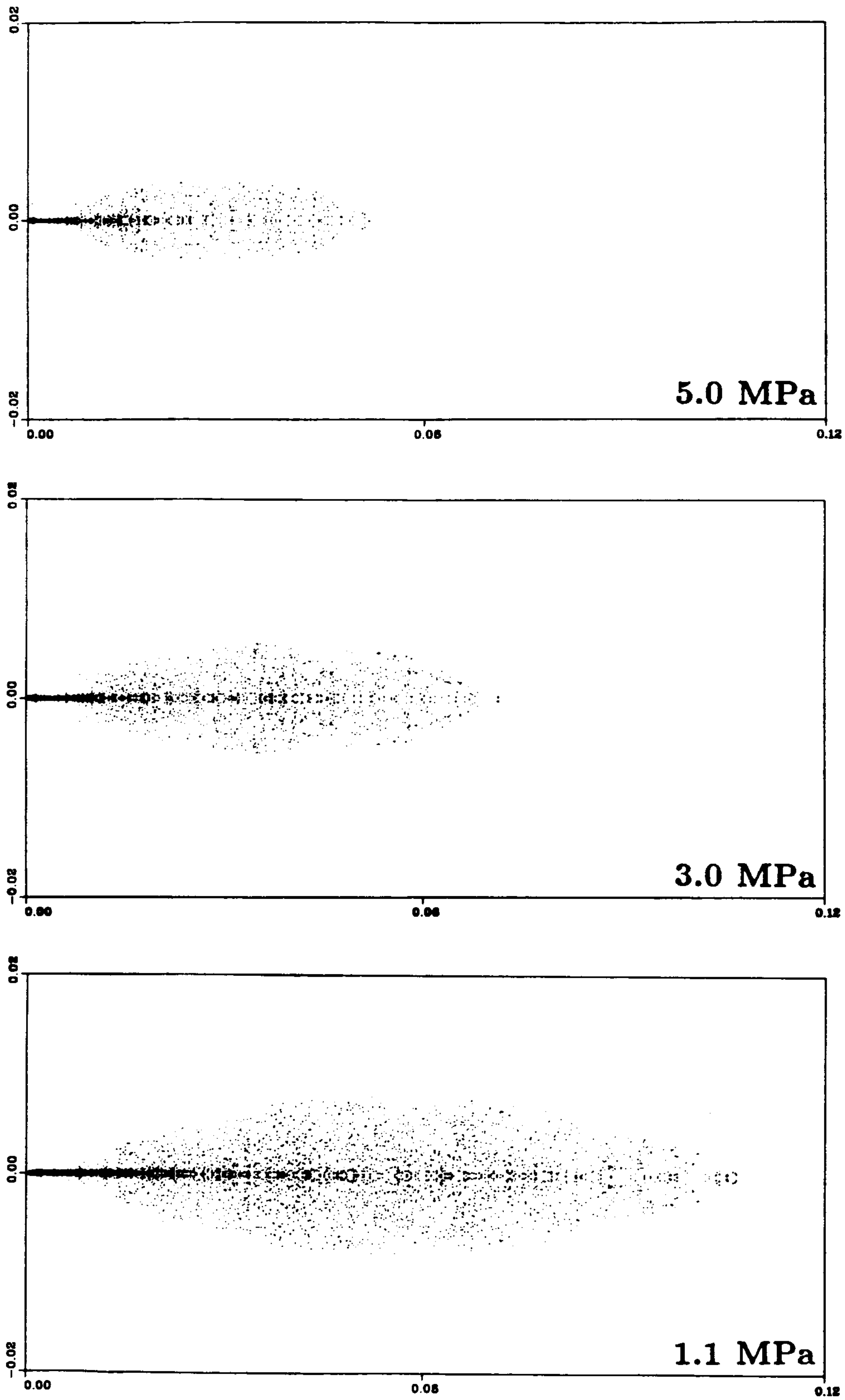
Figure 4.8.9 Effects of Upwing Damping on Stagnation Line Temperature Profiles

```
;------------------------- SOLID-CONE SPRAY ---------------------------
CONTROL  COMPRES
OMGD 1 PHI -1.00 OMGPHI 1.000 ERRCG 1.E-3 ERRM +1.E-7
NCRT 2  OMGF 1.0  IMON  10  JMON 10    MONU
;RESTART
GRID  NX 45  NY 25  AXISYM
      XDIR IST  1  IEND 2   DST -1.E-3 DEND  0.000    EXP  1.0
      XDIR IST  2  IEND 45  DST  0.00  DEND  0.12     EXP  1.3
      YDIR IST  1  IEND 10  DST  0.    DEND  5.E-3    EXP  1.0
      YDIR IST  10 IEND 25  DST  5.E-3 DEND  0.020    EXP  1.3
BOUND
       IST  1  IEND  45 JST  1 JEND  1  SYMMETRY
       IST  1  IEND  45 JST 25 JEND 25  OUTLET P 1.1E+6
       IST  1  IEND  1  JST  1 JEND 25  WALL
       IST  45 IEND  45 JST  1 JEND 25  OUTLET P 1.1E+6
PROPERTY  VISCOS  1.7E-5 DENGAS 12.36  UIN 0.00  TIN 298 PIN 1.1E+6
          YNIN 1.0 YFIN 0.0 YOIN 0.0
TURBULEN  TKIN 1.0E-3 TEIN 4.0E-4
SPRAY     SMR  150.E-6 CONST  DENPT 840  TKESW BREAKUP COLIDE
          CONE2  10  CONE1 0   MODUL  0
          IST  2 IEND  2 JST 2  JEND 2  NPTS 4
          VINJ 115.8 FLOWP 6.876E-3
          END
EVAP      TEMP 298  TABEL
SOLV      U  V   P  TK  TE  PATC
RUN       DT 10.0E-6   NSTEP 250  NPR1 10 NPR2 50
ENDJOB
```

Figure 4.9.1 Input deck for solid-cone spray

```
;------------------------- SOLID-CONE SPRAY ----------------------------
CONTROL  COMPRES
OMGD 1 PHI -1.00 OMGPHI 1.000 ERRCG 1.E-3 ERRM +1.E-7
NCRT 2  OMGF 1.0  IMON  10  JMON 10    MONU
;RESTART
GRID  NX 45  NY 25   AXISYM
      XDIR IST  1  IEND 2   DST -1.E-3 DEND  0.000    EXP  1.0
      XDIR IST  2  IEND 45  DST  0.00  DEND  0.12     EXP  1.3
      YDIR IST  1  IEND 10  DST  0.    DEND  5.E-3    EXP  1.0
      YDIR IST  10 IEND 25  DST  5.E-3 DEND  0.020    EXP  1.3
BOUND
       IST  1  IEND  45 JST  1 JEND  1  SYMMETRY
       IST  1  IEND  45 JST 25 JEND 25  OUTLET P 3.0E+6
       IST  1  IEND  1  JST  1 JEND 25  WALL
       IST  45 IEND  45 JST  1 JEND 25  OUTLET P 3.0E+6
PROPERTY  VISCOS  1.7E-5 DENGAS 33.70  UIN 0.00  TIN 298 PIN 3.0E+6
          YNIN 1.0 YFIN 0.0 YOIN 0.0
TURBULEN  TKIN 1.0E-3 TEIN 4.0E-4
SPRAY     SMR  150.E-6 CONST  DENPT 840  TKESW BREAKUP  COLIDE
          CONE2  10  CONE1 0   MODUL  0
          IST  2 IEND  2 JST 2  JEND 2  NPTS 5
          VINJ 102.5 FLOWP 6.090E-3
          END
EVAP      TEMP 298  TABEL
SOLV      U  V   P  TK  TE  PATC
RUN       DT 18.0E-6   NSTEP 250  NPR1 10 NPR2 10
ENDJOB
```

```
;------------------------- SOLID-CONE SPRAY ----------------------------
CONTROL  COMPRES
OMGD 1 PHI -1.00 OMGPHI 1.000 ERRCG 1.E-3 ERRM +1.E-7
NCRT 2  OMGF 1.0  IMON  10  JMON 10    MONU
;RESTART
GRID  NX 45  NY 25   AXISYM
      XDIR IST  1  IEND 2   DST -1.E-3 DEND  0.000    EXP  1.0
      XDIR IST  2  IEND 45  DST  0.00  DEND  0.12     EXP  1.3
      YDIR IST  1  IEND 10  DST  0.    DEND  5.E-3    EXP  1.0
      YDIR IST  10 IEND 25  DST  5.E-3 DEND  0.020    EXP  1.3
BOUND
       IST  1  IEND  45 JST  1 JEND  1  SYMMETRY
       IST  1  IEND  45 JST 25 JEND 25  OUTLET P 5.0E+6
       IST  1  IEND  1  JST  1 JEND 25  WALL
       IST  45 IEND  45 JST  1 JEND 25  OUTLET P 5.0E+6
PROPERTY  VISCOS  1.7E-5 DENGAS 56.17  UIN 0.00  TIN 298 PIN 5.0E+6
          YNIN 1.0 YFIN 0.0 YOIN 0.0
TURBULEN  TKIN 1.0E-3 TEIN 4.0E-4
SPRAY     SMR  150.E-6 CONST  DENPT 840  TKESW BREAKUP  COLIDE
          CONE2  10  CONE1 0   MODUL  0
          IST  2 IEND  2 JST 2  JEND 2  NPTS 5
          VINJ 86.41 FLOWP 5.130E-3
          END
EVAP      TEMP 298  TABEL
SOLV      U  V   P  TK  TE  PATC
RUN       DT 18.0E-6   NSTEP 250  NPR1 10 NPR2 10
ENDJOB
```


0.02
0.00
-0.02
5.0 MPa
0.00
0.06
0.12
0.02
0.00
-0.02
3.0 MPa
0.00
0.06
0.12
0.02
0.00
-0.02
1.1 MPa
0.00
0.06
0.12

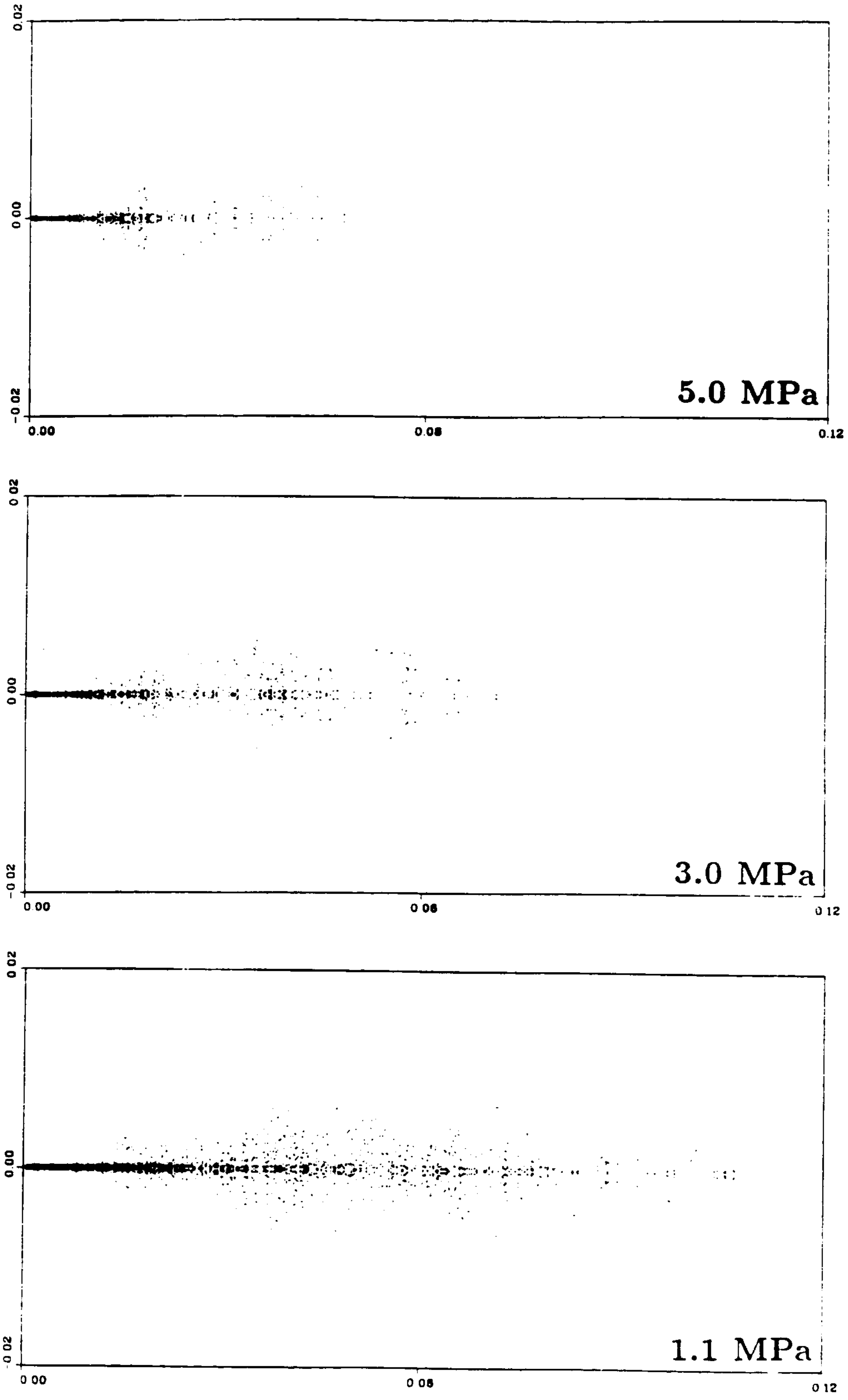


Figure 4.9.2 Spray parcel distribution in a solid-cone spray (t=1.0ms)

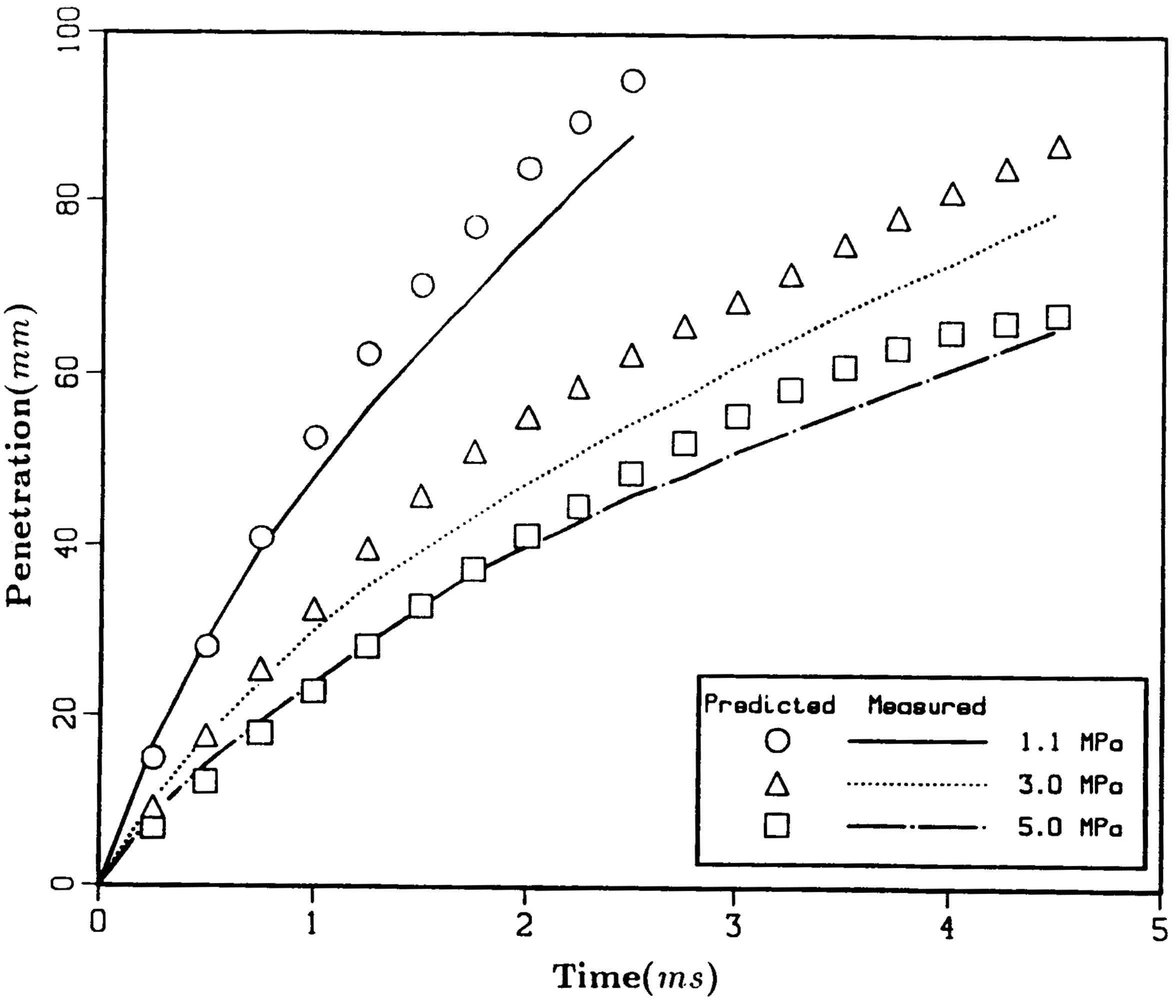


Figure 4.9.3 Spray tip penetration versus time

```
;------------------------- HOLLOW-CONE SPRAY ---------------------------
CONTROL  COMPRES
OMGD 1 PHI -1.00 OMGPHI 1.000 ERRCG 1.E-3 ERRM +1.E-7
NCRT 2  OMGF 1.0  IMON  10  JMON 10    MONU
;RESTART
GRID  NX 81  NY 41   AXISYM
      XDIR IST  1  IEND 81  DST  0.00  DEND  0.040    EXP  1.0
      YDIR IST  1  IEND 41  DST  0.00  DEND  0.020    EXP  1.0
BOUND
       IST    1 IEND   5 JST  1 JEND  3  BLOCK
       IST    6 IEND  81 JST  1 JEND  1  SYMMETRY
       IST    1 IEND   1 JST  1 JEND 41  WALL
       IST    1 IEND  81 JST 41 JEND 41  OUTLET P 0.55E+6
       IST   81 IEND  81 JST  1 JEND 41  OUTLET P 0.55E+6
PROPERTY  VISCOS  1.7E-5 DENGAS 6.360  UIN 0.00  TIN 300 PIN 0.55E+6
          YNIN 1.0 YFIN 0.0 YOIN 0.0
TURBULEN  TKIN 1.0E-3 TEIN 4.0E-4
SPRAY     SMR  30.E-6 CONST  DENPT 840  TKESW  BREAKUP COLIDE
          CONE1  30  CONE2 30  MODUL  0
          IST  6 IEND  6 JST 3  JEND 3  NPTS 5
          VINJ 60 FLOWP 5.980E-3
          END
EVAP      TEMP 300  TABEL
SOLV      U  V   P  TK  TE  PATC
RUN       DT  5.E-6   NSTEP 200  NPR1 10 NPR2 50
ENDJOB
```

Figure 4.10.1   Input deck for hollow-cone spray

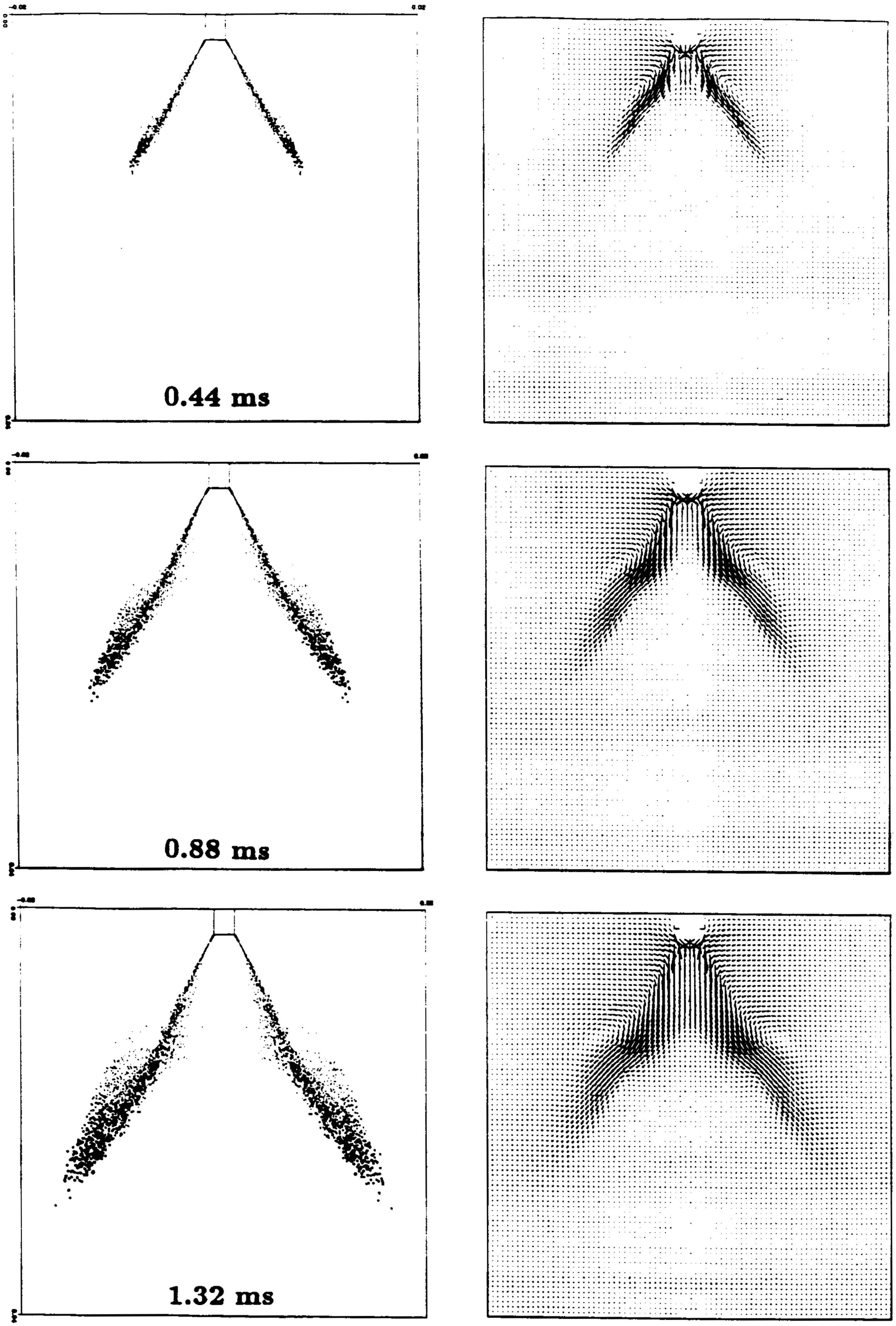


Figure 4.10.2 Spray Parcel distribution and velocity vectors in a hollow-cone spray (P=550 kPa)

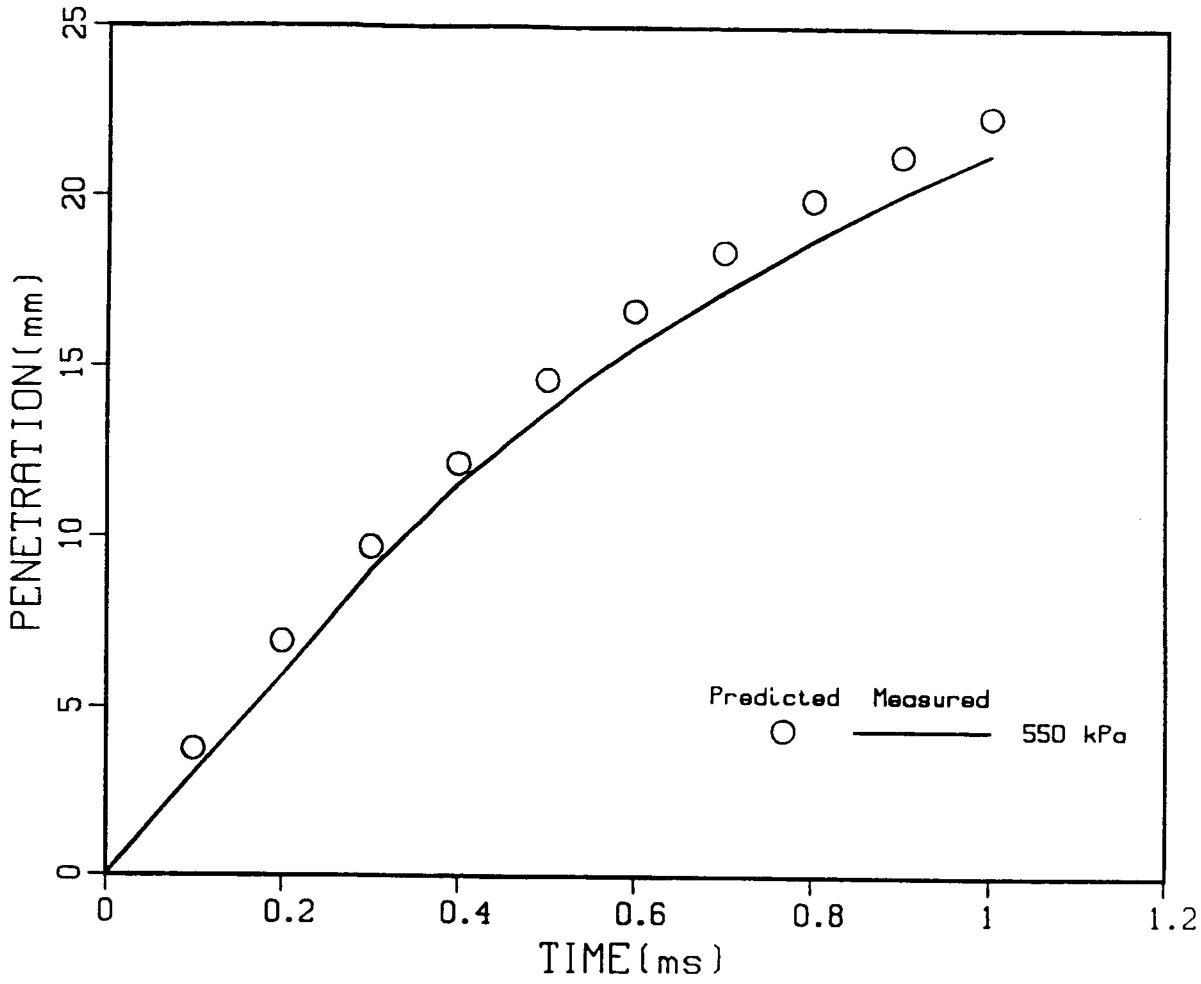


Figure 4.10.3 Spray tip penetration versus time

```
:------------------- HOLLOW-CONE SPRAY FLAME ------------------------
HYBRID  SWIRL  IMON 4   JMON 3   MONU  ERRCG 1.E-1 ERRM 1E-8
         IPF  2   JPF  2    PREF 1.0568E+5
RESTART
GRID  NX 61    NY 41       AXISYM
      XDIR IST  1 IEND 61 DST  0.00 DEND 2.000  EXP 1.3
      YDIR IST  1 IEND 41 DST   0   DEND 0.100  EXP 1.0
BOUND
          IST  1  IEND  61 JST  1 JEND   1  SYMMETRY
          IST  1  IEND  61 JST 41 JEND  41  WALL  Q 0
          IST  1  IEND   1 JST  1 JEND   8  WALL  Q 0
          IST  1  IEND   1 JST  9 JEND  19  INLET U 15.0 OMEGA 22.3
          TK 1.125  YFU 0.0  YOX 0.233  G 0.0
          IST  1  IEND   1 JST 20 JEND  41  WALL  Q 0
          IST 61  IEND  61 JST  1 JEND  41  OUTLET
PROPERTY  VISCOS  2.0E-5   TEMP 310  KGAS 0.026E+3 CPGAS 1000
          DENGAS 1.183
TURBULEN  CT1 1.44  CT2 1.92  CMU 0.09  SME 1.3  SMK 1.0
          TKIN 1.125
SPRAY     SMR  +6.35E-5  DENPT 770 TEMP 310  EVAP
          MODUL  0
          IST  1 IEND  1 JST 1 JEND 1 NPTS 15 VINJ 11. FLOWP 4.89E-3
SOLV      U    V    P   TK  TE  SW  FU  PATC  G
RUN       DT 2.0E-4   NTIME 250 NPR1 10 NPR2 10
ENDJOB
```

Figure 4.11.1   Input deck for hollow-cone spray flame

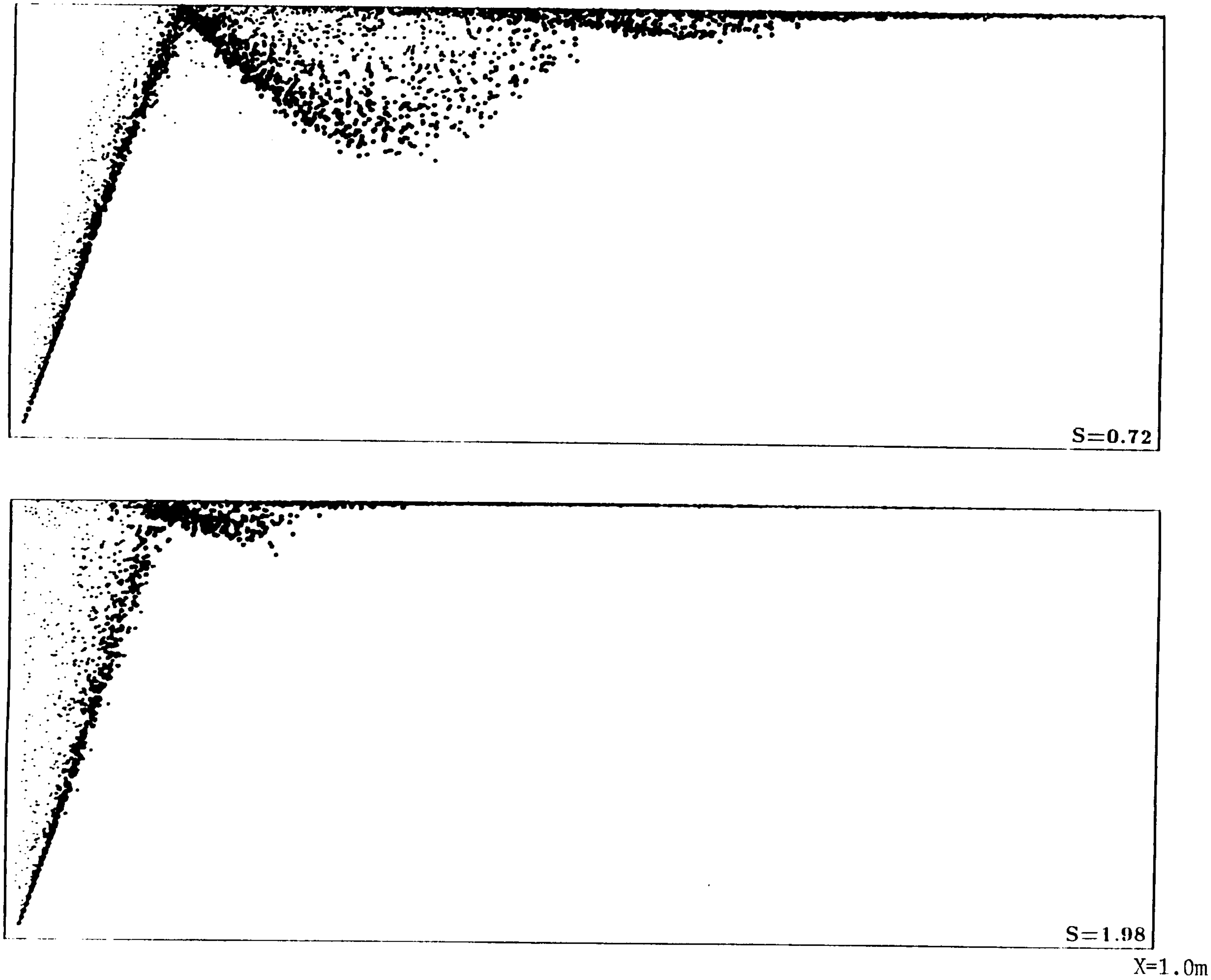


Figure 4.11.2 Droplet trajectories in kerosene spray flame fields

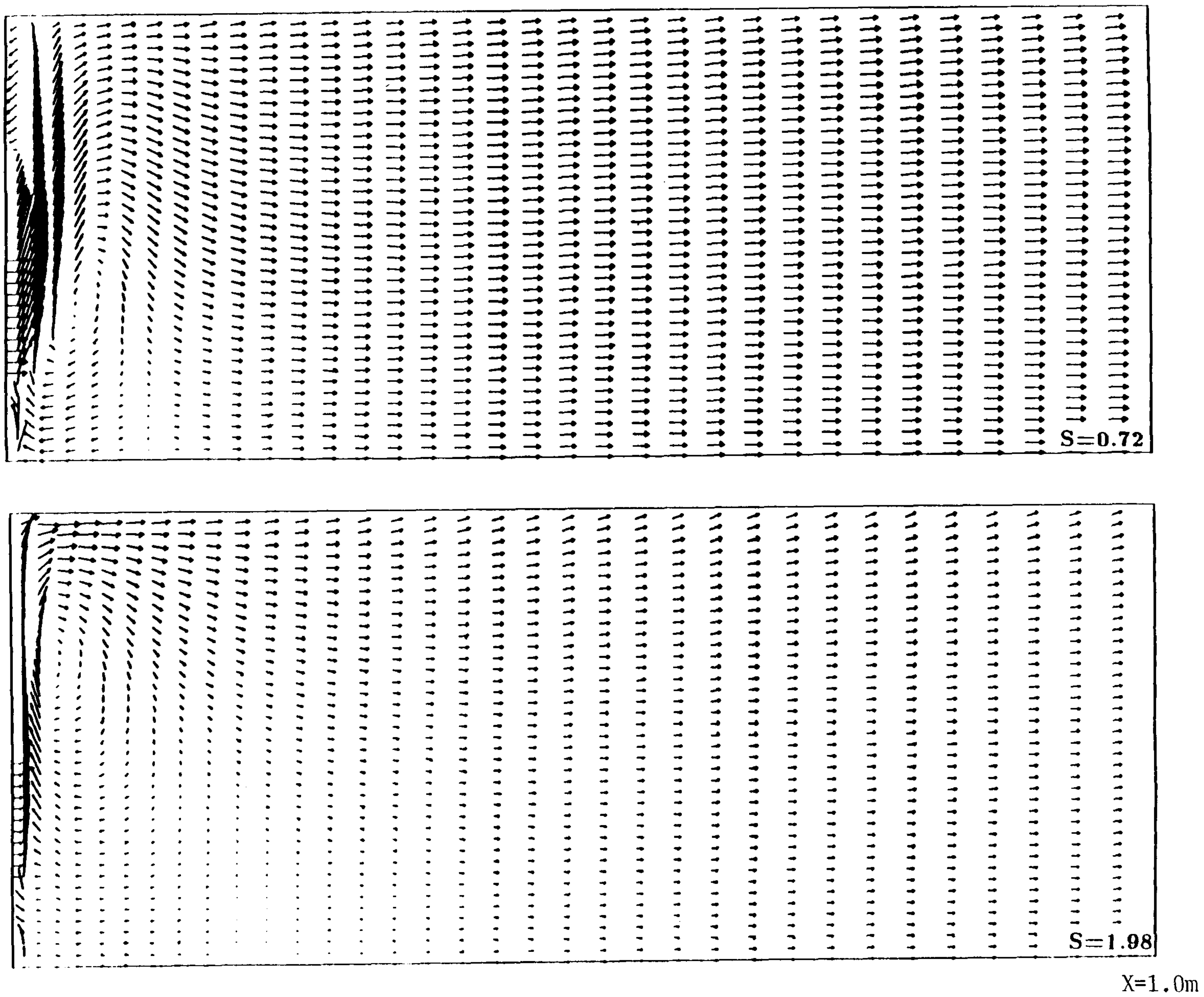


Figure 4.11.3 Velocity vectors in kerosene spray flame fields

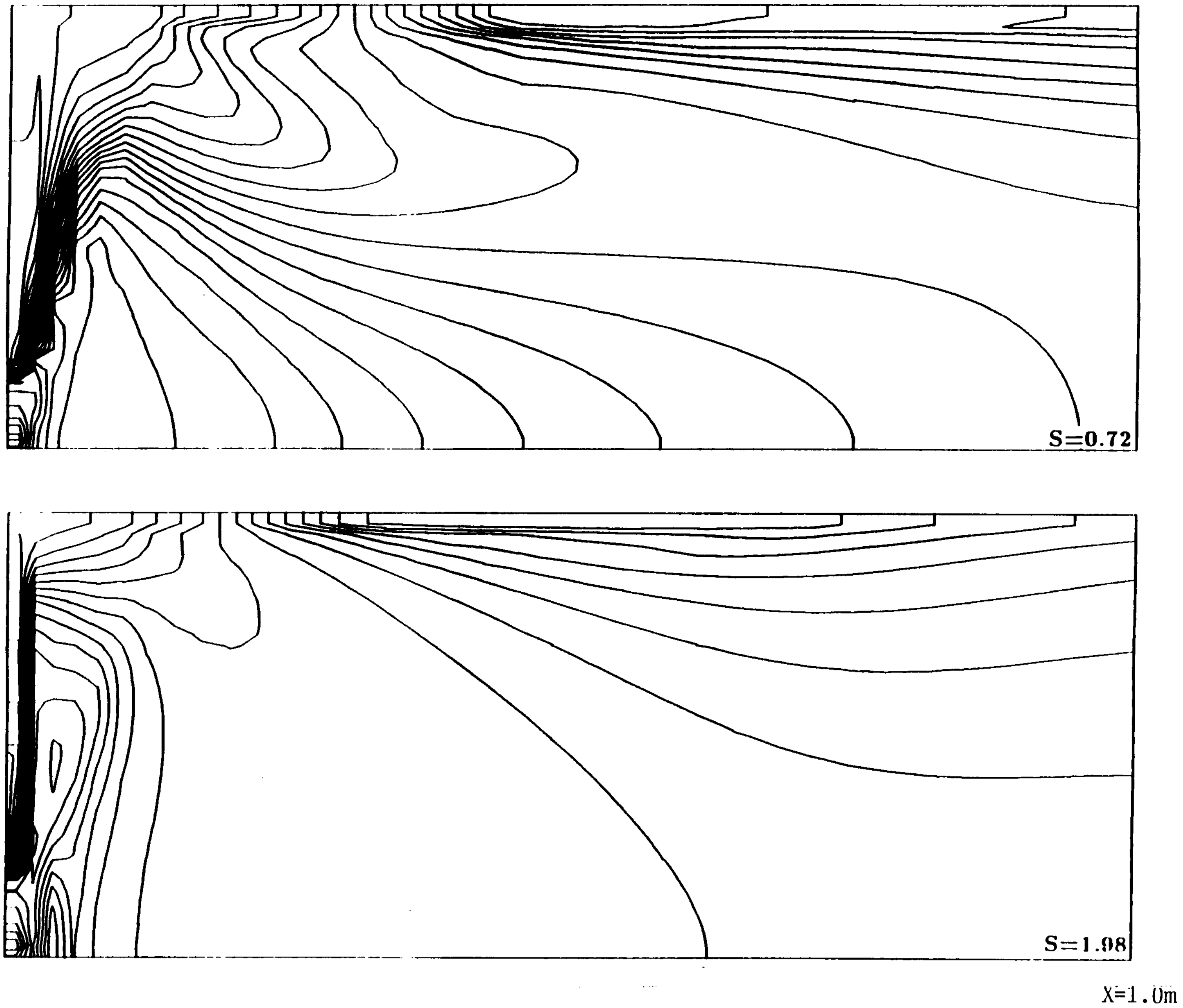


Figure 4.11.4 Temperature contours in kerosene spray flame fields

APPENDIX A

List of FORTRAN Symbols

AE( )	discretization coefficient of east point
AIRDIF	molecular diffusivity of fuel vapor in ambient medium
	is AIRDIF*TEMP**EXPDIF/RHO
AIRLA1	molecular heat conduction coefficient of ambient medium
AIRLA2	is AIRLA1*TEMP**1.5/(TEMP+AIRLA2)
AIRMU1	molecular viscosity of ambient medium
AIRMU2	is AIRMU1*TEMP**1.5/(TEMP+AIRMU2)
AK( )	heat conductivity k
AKG	constant heat conductivity
AKPAS( )	heat conductivity at patch
ALF( )	declination angle in radian between grid line and boundary
AM( )	Mach number
AMMR	droplet mass median radius
AMW( )	molecular weighting factor
AN( )	discretization coefficient of north point
ANE( )	discretization coefficient of north east point
ANW( )	discretization coefficient of north west point
AP( )	discretization coefficient of main point
AP0( )	coefficient of CG solver
AS( )	discretization coefficient of south point
ASE( )	discretization coefficient of south east point
ASW( )	discretization coefficient of south west point
AW( )	discretization coefficient of west point
B0( )	coefficient of CG solver
BIG	a very large real number
BREAKUP	switch of BREAKUP model(1 - on, 0 - off)
CMU	turbulence model constant
CONE1	fuel injector spray cone angle
CONE2	(CONE2>CONE1)
CP( )	specific heat
CPG	constant specific heat
CPPAS( )	specific heat at patch
CSUBK	constant of BREAKUP model
CSUBMU	constant of BREAKUP model
CSX( )	$\partial\xi/\partial x$
CSY( )	$\partial\xi/\partial y$
CTX( )	$\partial\eta/\partial x$
CTY( )	$\partial\eta/\partial y$
CT1	turbulence model constant
CT2	turbulence model constant
D0( )	coefficient of CG solver
DELT( )	first grid cell size of non-uniform grid section
DEN( )	density
DEN0( )	density of last time step

DENA( )	density of the time step before the last
DENGAS	constant density
DENPAS( )	density at patch
DET( )	Jacobian of coordinate tranformation
DLEND( )	end position of non-uniform grid section
DLH( )	distance between boundary and first grid point from wall
DLST( )	start position of non-uniform grid section
DMAX	control criterion in CG solver
DMTOT( )	particle contribution to cell mass
DPX( )	pressure gradient in X direction
DPY( )	pressure gradient in Y direction
DSIEP( )	particle contribution to cell specific internal energy
DSX( )	distance between grid point in X direction
DSY( )	distance between grid point in Y direction
DT	time step
DTEPS( )	particle contribution to cell turbulence ε
DTKEP( )	particle contribution to cell turbulence k
DTMAX	maximum time step
DTMIN	minimum time step
DU( )	connection coefficients of pressure correction and velocity correction
DV( )	
EK( )	a conversion of HK array
ELIQ( )	a conversion of HLATO array
ENTHO( )	initial cell enthalpy in EVAP
EPS( )	ε of turbulence model
ERRCG	convergence criterion for conjugate gradient matrix solver
ERRM	convergence criterion for steady state solution
EVAPM	particle evaporation mass in domain
EVAPP	switch of evaporation model
EXPDIF	(see AIRDIF)
EXPQ	constant for particle size distribution
FERR( )	relative error of each time step
FEXP( )	stretching factor of non-uniform grid
FINP( )	interpolation function for boundary point
FLOWP( )	injected particle mass flow rate
FN( )	array to store all the main variables
FP( )	temporary array
GAM( )	Γ of differential equation
GAMF	constant for particle size distribution
GAMI	constant ratio of specific heat
GAMMA( )	ratio of specific heat
GENT( )	turbulence generation term
GI( )	pressure damping term used in scheme
GJ( )	pressure damping term used in scheme
GMW	molecular weighting factor
GPI( )	pressure damping term used in face velocity
GPJ( )	pressure damping term used in face velocity
GX	gravity in X direction (used in particle phase)
GY	gravity in Y direction (used in particle phase)

GZ	gravity in Z direction (used in particle phase)
HO( )	coefficient of CG solver
HB( )	coefficient of CG solver
HK( )	enthalpy of species at T=100(N-1)
HLATO( )	latent heat of liquid
HPAS( )	enthalpy at patch
HPTC( )	for generating particle size
HPU( )	operator for finite-difference representation of
HPV( )	momentum equation
HSTAG	total enthalpy
HTFORM( )	heat of formation of species
IBC( )	I index of boundary point
IDBC( )	type of boundary condition
IDFI( )	flag of variables specified at boundary
IDSP( )	identify species
IEND( )	patch ending point
ILEND( )	end point of non-uniform grid section
ILST( )	start point of non-uniform grid section
IMON	monitor point I index
INCOMP	switch of compressible (1 - incomp. 0 - comp.)
INEB( )	specify first grid point from wall
INX( )	relative I index of first grid point from wall
INY( )	relative J index of first grid point from wall
IPEND( )	particle injection position,ending
IPST( )	particle injection position,starting
IPTC( )	particle index at cell
IST( )	patch start point in I direction
JBC( )	J index of boundary point
JEND( )	patch ending point in J direction
JMON	J index of monitor point
JPEND( )	particle injection position,ending
JPST( )	particle injection position,starting
JPTC( )	particle index at cell
JST( )	patch start point in J direction
KOLIDE	switch for particle collision model
KSIZE	specify particle size distribution (0 - constant, 1 - modified Rosin-Rammler, 2 - Chi-Squared)
LDBC( )	boundary condition type of patch
LF	fuel variable index in FN( )
LH	index for enthalpy in FN( )
LO	oxygen index in FN( )
LP	index for pressure in FN( )
LPATC	switch for particle tracking
LPFAC	switch for face velocity printing
LPGEO	switch for geometry parameter printing
LPT1	printing flag
LPT2	printing flag
LREST	switch of restart
LS( )	index of variable need to be solved

LSTOP	identification of stop of time marching
LTE	index of turbulence model ε in FN( )
LTK	index of turbulence model k in FN( )
LU	index of U velocity in FN( )
LV	index of V velocity in FN( )
LW	index of swirl velocity in FN( )
MBK	maximum dimension of patch related array
MBL( )	identification of block point
MNP	maximum dimension of particle arrays
MODUL	particle turbulence modulation models
MPM	maximum dimension of boundary related arrays
MSP	maximum number of species
MSPI	maximum dimension of 2-D species arrays
MVAP	maximum dimension of fuel property variables
MX	maximum dimension of I direction
MY	maximum dimension of J direction
NAS	switch of axis symmetry
NBC	number of boundary point
NCHEM	switch of chemical reaction
NCG( )	number of CG iteration used
NCGM	maximum CG solver iteration
NCRT	number of pressure correction steps
NDIR( )	direction of non-uniform grid
NEX	index of example case
NF	index of variable in FN( )
NFL	switch of laminar flow
NFMAX	maximum index of FN( )
NFMN	index of monitor variable
NGRD	number of grid section
NP	number of patch
NPC( )	particle number in a certain grid point
NPR1	output frequency
NPR2	output frequency
NPTM	current particle number in domain
NPTS( )	particle number injected per time step
NSP	number of species to be solved
NSTEP	maximum time marching step
NSW	switch of swirl flow
NT	number of time marching steps
NUMNOZ	number of nozzles
NX	number of point in I direction
NX1	NX-1
NX2	NX-2
NY	number of point in J direction
NY1	NY-1
NY2	NY-2
OMG( )	rotation per second
OMGD	weighting factor for continuity density convection term
OMGF	face velocity interpolation weighting factor

OMGPHI	weighting factor of upwinding in Chakravarthy and Osher scheme
OMGT	supersonic temperature field relaxation
OSCILO( )	initial droplet oscillation frequency at injector in BREAKUP
P( )	pressure
P0( )	coefficient of CG solver
P1( )	coefficient of CG solver
PARTN( )	particle number in the computational parcel
PCRIT	liquid phase critical pressure
PFO( )	pressure of last time step
PHI	parameter of Chakravarthy and Osher scheme
PI	π
PI403R	4/3*PI*RHOP
PIN	initial pressure
PPAS( )	pressure at patch
PRG	laminar Prandtl number
PRT	turbulent Prandtl number
PSFA	constant for liquid phase vapor pressure
PSFB	constant for liquid phase vapor pressure
PSTAG	stagnation pressure
PTEMP	initial particle temperature
PVAP( )	liquid phase vapor pressure
QPAS( )	heat flux at patch
R0( )	coefficient of CG solver
R1( )	coefficient of CG solver
RADP( )	particle size before evaporation
RADPP( )	particle size after evaporation
RCON( )	gas constant
RCONI	initial gas constant
RELVEL( )	particle-gas relative velocity
RERF( )	inverse error function
RHO( )	continuous phase density = DEN( )
RHOI( )	initial species density
RHOP	particle density
RMW( )	1/AMW
RPMAX	maximum particle size
RU( )	particle contribution to cell momentum
RV( )	particle contribution to cell momentum
RW( )	particle contribution to cell momentum
SC( )	source term of discretization equation
SCALE	length scale for initial turbulence ε
SIE( )	enthalpy = FN(I,J,LH)
SIXTH	1/6
SMALL	a very small real number
SMAX	maximum mass residual in continuity
SME	turbulence model constant
SMK	turbulence model constant
SMR	Sauter mean radius

SMT	turbulence model constant
SP( )	discretization coefficient of source term
SPDRAG( )	particle drag
SPKT( )	turbulence source term splitting
SPMTIL( )	explicit portion of contribution to mass diffusion
SSUM	summation of pressure correction equation source term
ST350	calculation of particle surface tension
STB	calculation of particle surface tension
STM	calculation of particle surface tension
SUVW( )	particle coupling term for momentum exchange
SWC( )	swirl velocity source term
SWP( )	swirl velocity source term
TO( )	temperature of last time step
TABEL	flag for property using table or correlation
TBN	liquid phase normal boiling temperature
TCRIT	liquid phase critical temperature
TEIN	initial turbulence ε
TEMP( )	temperature = TN( )
TEPAS( )	turbulence ε at patch
THIRD	1/3
TIMAX	maximum time step
TIME	current time marching
TIN	initial temperature
TKE( )	turbulence k
TKESW	flag for particle turbulent dispersion
TKIN	initial turbulence k
TKPAS( )	turbulence k at patch
TN( )	continuous phase temperature
TOTCM( )	total cell mass in EVAP
TOTH( )	total cell enthalpy in EVAP
TP( )	particle temperature
TPAS( )	temperature at patch
TS( )	intermediate step temperature
TSTAG	stagnation temperature
TURBT( )	turbulence correlation time for particle diffusion
TWOTHD	2/3
U( )	U velocity
UCF( )	face velocity U
UCFO( )	last time step face velocity U
UIN	initial U velocity
UP( )	particle velocity U
UPAS( )	U velocity at patch
US2( )	temporary array
UTRB( )	gas phase turbulent velocity fluctuation
V( )	V velocity
VAPM( )	vapor mass in cell
VCF( )	face velocity V
VCFO( )	last time step face velocity V
VIN	initial V velocity

VINJ( )	particle injection velocity
VISCOS	viscosity
VISLIQ( )	liquid phase viscosity
VISM( )	laminar viscosity
VIST( )	turbulent viscosity
VOL( )	volume of cell
VP( )	particle velocity V
VPAS( )	V velocity at patch
VS2( )	temporary array
VTRB( )	gas phase turbulent velocity fluctuation
W( )	swirl velocity
W0( )	swirl velocity of last time step
WM( )	molecular weighting factor
WP( )	particle velocity
WTRB( )	gas phase turbulent velocity fluctuation
X( )	grid point X position
XBC( )	X projection of the first grid point away from wall
XLEN	X direction expansion of rectangular grid
XP( )	particle location
Y( )	grid point Y position
YBC( )	Y projection of the first grid point away from wall
YLEN	Y direction expansion of rectangular grid
YMAS( )	mass fraction
YOP( )	droplet deviation from spheroidicity
YOPDOT( )	time rate of change of YOP
YP( )	particle location
ZP( )	particle location

APPENDIX C

Preprinted from the Int. J. Numerical Method of Fluids, 1992.

AIAA-91-0286
A Novel Gas-Droplet Numerical Method for Spray Combustion

C. P. Chen
H. M. Shang
Y. Jiang
Department of Chemical Engineering
University of Alabama in Huntsville

29th Aerospace Sciences Meeting

January 7-10, 1991/Reno, Nevada

A NOVEL GAS-DROPLET NUMERICAL METHOD FOR SPRAY COMBUSTION

C. P. Chen
H. M. Shang
Y. Jiang
Department of Chemical Engineering
University of Alabama in Huntsville
Huntsville, AL 35899

ABSTRACT

This paper presents a non-iterative numerical technique for computing time-dependent gas-droplet flows. The method is a fully-interacting combination of Eulerian fluid and Lagrangian particle calculation. The interaction calculations between the two phases are formulated on a pressure-velocity coupling procedure based on the operator-splitting technique. This procedure eliminates the global iterations required in the conventional particle-source- in-cell (PSIC) procedure. Turbulent dispersion calculations are treated by a stochastic procedure. Numerical calculations and comparisons with available experimental data, as well as efficiency assessments are given for some sprays typical of spray combustion applications.

NOMENCLATURE

C_D	Drag coefficient
C_μ	Model constant for the $k-\epsilon$ model, $= 0.09$
d_p	Droplet diameter for computational particle p
f	$C_D Re_p/24$ $= 1$, for $Re_p \leq 1$ $= 1 + 0.15\ Re_p^{0.687}$, for $Re_p > 1$
F_i	Interaction force due to droplets
F_{pi}	Particle drag force
g_i	Gravity vector
k	Kinetic energy
N_p	Number of droplets for each computational particle p
NP	Number of computational particles
P	Mean pressure
r_p	Droplet radius
Re_p	Particle Reynolds number $= \frac{\rho d_p \lvert U_i + u'_i - v_i \rvert}{\mu}$
S_i	Source terms in the momentum equation
t_e	Eddy life time
t_{int}	Droplet-turbulence interaction time
t_{tr}	Particle transit time
t_*	Particle relaxation time $= \rho_d d_p^2/18\mu$
u_i	Instantaneous velocity for gases $= U_i$(mean) + u'_i(fluctuation)
v_i	Instantaneous velocity for droplets
τ	Effective particle relaxation time
ρ	Fluid (gas) density
ρ_d	Droplet density
μ	Gas viscosity
ϵ	Dissipate rate of turbulent kinetic energy

I. INTRODUCTION

Numerical modeling of two-phase, turbulent reacting flows has practical applications in the development and design of many power generating devices such as internal combustion engines and liquid-rocket engines. In these flows, two-way coupling in terms of momentum, heat and mass transfer between underlying gas turbulence and dispersed spray droplets play one of the most important physical role of mixing and consequent spray combustion. In the last two decades computational techniques have been developed to characterize the dispersion of spray droplets in a gas and the influences of droplets on the gas dynamics. Generally, there are two approaches commonly used to predict gas-droplet flows. One, called the "two-fluid" model or Eulerian approach. In this approach, the effect of two-way coupling is incorporated as extra source terms in the continuum equations for both phases. The advantage of using a continuum approach is its relatively computational efficiency especially for mono-dispersed systems.

Another approach. the so called "tracking" or Lagrangian approach. treats the particles as discrete entities in a turbulent flow field and their trajectories are calculated. This approach has the flexibility of handling poly-dispersed spray and the two-way couplings are usually accomplished through the particle- source-in-cell (PSIC) technique with exchange of momentum, heat and mass between the two phases. Both approaches have been studied extensively [1,2] and comparative performances of these two approaches have also being investigated recently [3,4]. For typical spray combustion applications in which dense spray effects such as droplet collision, breakup and coalescence are important and drop dispersions are characterized by a non-uniform particle size distribution , the discrete particle approach is more convenient for representing the poly-dispersed spray.

In calculating turbulent gas-droplet flows, the most common discrete particle method is the stochastic separated flow approach as first described by [5] utilizing the method of Crowe et al. [6]. In this method the liquid spray droplets are represented by a finite number of computational parcels and a random sampling technique is entailed for instantaneous gas flow properties based on a specified turbulence model and the resulting fluctuations are used in a Lagrangian computation for parcel motion. This method has been used primely for statistically steady flows in which the global iterations [7] between the continuous phase solver and particle equation of motion are invoked [8,9]. For transient problems, this global iteration at each time step can introduce such excessive computational requirements that numerical simulation becomes impractical. Dukowicz [10] has introduced a time-splitting method to couple the gas-particle interactions in a transient calculation. The numerical scheme used in his method is based on the SOLA code which utilizes a pressure substitution scheme. A similar time-splitting method has been used very recently by [11] in a pressure correction scheme for steady- state calculations and by [12] in a density-based scheme.

The main effort of this paper is to present a numerical method for coupling the gas-droplet interactions using a pressure correction scheme. This method utilizes the operator-splitting technique in deriving a predictor multi-corrector sequence which eliminates the global iteration between the two- phases at each time step. We have found that this method is efficient and that the required number of computational parcels to achieve satisfactory accuracy is also not excessive. In the following sections, formulations and validations of this method are presented.

II. NUMERICAL MODEL

The gas flow was formulated using the Eulerian conservation equations of mass and momentum. The spray is described by the discrete particle method formulated on a Lagrangian frame. The spray is assumed to be sufficiently dispersed (no collision between droplets), and for simplicity, the gas flow is assumed to be close to incompressible. The governing equations are: Gas phase

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho U_i) = \dot{S}^m \qquad (1)$$

$$\frac{\partial \rho U_i}{\partial t} + \frac{\partial}{\partial x_i}(\rho U_i) = -\frac{\partial P}{\partial x_j} - \frac{\partial}{\partial x_j}(\Gamma_{ij}) + \overline{F}_i \qquad (2)$$

Here all variables are ensemble-averaged mean quantities, and

$$\Gamma_{ij} = -(\mu + \mu_t)[\frac{1}{2}(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i})] + \frac{2}{3}\delta_{ij}k \qquad (3)$$

in which μ_t is the eddy viscosity. Particle phase

$$\frac{dx_i}{dt} = v_i \qquad (4)$$

$$\frac{dv_i}{dt} = F_{pi} + g_i \qquad (5)$$

Since the formulation here is essentially a statistical approach, each computational parcel represents a large number of droplets having equal location , velocity, size and temperature. The two-way coupling between the two phases is accounted for by the interaction terms, where

$$F_{pi} = \frac{U_i + u_i' - v_i}{\tau} \qquad (6)$$

$$\dot{S}^m = \sum_{p=1}^{NP} N_p \dot{m}_{ev,p}/dV \qquad (7),$$

for evaporating spray [13];

$$F_i = \sum_{p=1}^{NP} [N_p \dot{m}_{ev,p}(v_i)_p - \frac{4}{3}\pi \rho_d r_p^3 N_p (\frac{dv_i}{dt})_p]/dV \qquad (8),$$

in which dV denotes the computational cell and the effective relaxation time $\tau = t_*/f$, $t_* = \rho_d d_p^2/18$ and $f = C_D Re_p/24$.

It suffices here to illustrate the calculation method for non-evaporating case ($\dot{m}_{ev,p} = 0$). The method is based on the operator splitting technique attempting to reach accurate transient solution after prescribed predictor-corrector steps for each time-marching step. The generalization of this operator-splliting technique for deriving pressure-correction equations suitable for all speed flows is described separately [14]. We focus in this paper on the coupling of the dispersed phase in the solution procedure.

Discretization of the gas phase governing equation uses the finite volume approach. Differencing in the temporal domain employs the implicit Euler Scheme. All the dependent and independent variables are stored at the same grid location and the variables at the finite control volume boundaries are interpolated between adjacent grid points. The discretizations have been performed on a general non-orthogonal curvilinear coordinate system with a second order upwind scheme for convection terms and the central differencing scheme for diffusion terms [15].

For incompressible gas flows considered here, the pressure-velocity coupling between the momentum and continuity equation is an important issue since density is constant. In the pressure-correction method, the derivation of the pressure equation which includes the effect of droplets plays a key role in determining whether the velocity field satisfies the local mass continuity equation. In this study, a non-iterative operator-splitting algorithm following the spirit of [16,14] is used to derive a predictor-corrector sequence. We seek the finite difference form of the governing equations (2) and (5) as follow :

$$(\frac{\rho}{\Delta t} - A_o)U_i^{n+1} = H'(U_i^{n+1}) - \Delta_i P^{n+1} + S_i + \overline{F}_i^{n+1} \quad (9)$$

and

$$\frac{v_i^{n+1} - v_i^n}{\Delta t} = \frac{U_i^{n+1} + u_i' - v_i^{n+1}}{\tau^{**}} + g_i \quad (10)$$

The effective relaxation time scale τ is evaluated at the second corrector level (**), to be defined later. The superscripts n and $n+1$ denote time level t^n and t^{n+1} respectively. Operator A_o and $H'(\ )$ are constructed from the second-order upwind scheme for the convection terms and the central differencing scheme for the diffusion terms, and S_i is the source term associated with the Cartesian mean velocity component [14, 15]. It has been shown in [10] that the ensemble averaged interaction term $\overline{F}_i$ can be replaced by volume averaging. We split this term as follow :

$$\overline{F}_i^{n+1} = -S_u^{**}U_i^{n+1} + R_u^{**} \quad (11)$$

in which S_u^{**} and R_u^{**} are obtained by rearranging equation (10) :

$$S_u^{**} = \frac{1}{dV}\sum_p^{NP} N_p m_p/(\Delta t + \tau_p^{**}) \quad (12)$$

$$R_u^{**} = \frac{1}{dV}\sum_p^{NP} N_p m_p/(\Delta t + \tau_p^{**})(v_i^n - u_i' + g_i\Delta t)_p \quad (13)$$

and $m_p = \frac{4}{3}\pi r_p^3 \rho_d$ is the particle mass. The parameter S_u^{**} and R_u^{**} are momentum control volume quantities depending on available particle information at second corrector level to be discussed later.

By operator-splitting method, we divide the predictor-corrector procedure as following :

Predictor step :

$$(\frac{\rho}{\Delta t} - A_o)U_i^* = H'(U_i^*) - \Delta_i P^n + S_i - S_u^n U_i^* + R_u^n \quad (14)$$

$$\frac{v_i^* - v_i^n}{\Delta t} = g_i + \frac{U_i^* + (u_i')^n - v_i^*}{\tau^n} \quad (15)$$

The quantities S_u^n, R_u^n are determined from the existing flow fields. These values of U_i^* and v_i^* are used to evaluate τ^*,S_u^* and R_u^* such that a second approximation to the gas velocities can be performed :

$$(\frac{\rho}{\Delta t} - A_o)U_i^{*T} = H'(U_i^*) - \Delta_i P^n + S_i - S_i^* U_i^{*T} + R_u^* \quad (16)$$

By subtracting eq.(14) from eq.(16), we also obtain the velocity correction equation :

$$(\frac{\rho}{\Delta t} - A_o)(U_i^{*T} - U_i^*) = -S_u^* U_i^{*T} + S_u^n U_i^* + R_u^* - R_u^n \quad (17)$$

For the first corrector step, the momentum equation is approximated by

$$(\frac{\rho}{\Delta t} - A_o)U_i^{**} = H'(U_i^{*T}) - \Delta_i P^* + S_i - S_u^* U_i^{**} + R_u^* \quad (18)$$

By subtracting eq.(17) from eq.(18)

$$(\frac{\rho}{\Delta t} - A_o + S_u^*)(U_i^{**} - U_i^{*T}) = H'(U_i^{*T} - U_i^*)$$
$$-\Delta_i(P^* - P^n) \quad (19)$$

Taking divergence of eq.(19) and invoke continuity ($\nabla U_i^{**} = 0$), the pressure correction equation is obtained :

$$\Delta_i[D_u^* \Delta_i(P^* - P^n)] = \Delta_i U_i^{*T} + \Delta_i[D_u^* H'(U_i^{*T} - U_i^*)] \quad (20)$$

Here we have used the short notation $D_u^* = (\frac{\rho}{\Delta t} - A_o + S_u^*)^{-1}$. The particle momentum at this level is then

$$\frac{v_i^{**} - v_i^n}{\Delta t} = g_i + \frac{U_i^{**} + (u_i')^n - v_i^{**}}{\tau^*} \quad (21)$$

The values obtained at this level are used to calculate τ^{**},S_u^{**} , R_u^{**} and further update the velocities for gas phase :

$$(\frac{\rho}{\Delta t} - A_o)U_i^{**T} = H'(U_i^{*T}) - \Delta_i P^* + S_i - S_u^{**} U_i^{**T} + R_u^{**} \quad (22)$$

or, the velocity correction equation (eq.(22) - eq.(18)) :

$$(\frac{\rho}{\Delta t} - A_o)(U_i^{**T} - U_i^{**}) = -S_u^{**} U_i^{**T} + S_u^* U_i^{**} + R_u^{**} - R_u^* \quad (23)$$

At this stage, the mean velocity field satisfies the continuity constraint. To further satisfy the momentum conservation, a second corrector step is used :

$$(\frac{\rho}{\Delta t} - A_o)U_i^{***} = H'(U_i^{**T}) - \Delta_i P^{**} + S_i - S_u^{**}U_i^{***} + R_u^{**} \qquad (24)$$

By subtracting (22) from (24), taking divergence and invoke continuity again. we obtain the following pressure correction and velocity correction equations :

$$\Delta_i[D_u^{**}\Delta_i(P^{**} - P^*)] = \Delta_i U_i^{**T} + \Delta_i[D_u^{**}H'(U_i^{**T} - U_i^{*T})] \qquad (25)$$

$$(\frac{\rho}{\Delta t} - A_o + S_u^{**})(U_i^{***} - U_i^{**T}) = H'(U_i^{**T} - U_i^{*T})$$

$$-\Delta_i(P^{**} - P^*) \qquad (26)$$

Following [16], it can be shown that the errors introduced by the operator- splitting procedure is less than the truncation errors of the finite difference scheme used in the governing equations (9) and (10). Note that the effective relaxation time τ depends on the drag function which should contain the effects of turbulence. We therefore calculate $(u_i')^*$ at this stage by a stochastic method with the $k-\epsilon$ turbulence model [3,13]. A one-predictor (implicit)/one-correction (explicit) procedure for k and ϵ equations suggested by Issa [11] (see also [14]) has been used in this study. We then let U_i^{***} and P^{**} be the value at t^{n+1} level and add the $(u_i')^*$ to update the final time level particle velocities using the equation :

$$\frac{v_i^{n+1} - v_i^n}{\Delta t} = \frac{U_i^{n+1} + (u_i')^* - v_i^{n+1}}{\tau^{**}} + g_i \qquad (27)$$

This brings all variables to the new time level. The time is then incremented and the new predictor-corrector repeated with the new velocities. This algorithm is used for simulation of transient phenomena. If only a statistically steady solution is desired, then the time steps for gas phase and particle phase can be made unequal ; also the U_i^{*T} and U_i^{**T} calculation steps can be neglected.

III. NUMERICAL APPLICATIONS

The above mentioned procedures have been coded into the MAST (Multiphase All- Speed Transient Flow Solver) program for various two-phase flow calculations. We first present the solid particle dispersion calculation in nearly-homogenous turbulence to calibrate the stochastic simulation of particle-turbulence interactions.

Discrete Phase Turbulent Dispersion

The stochastic technique used for modeling the discrete phase turbulent dispersion is similar to the methods used by, for example, Dukowicz [10], Gosman and Ioannides [5], and Shuen et. al. [17]. The continuous phase turbulence is assumed to be isotropic, and the random turbulent velocity component u_i' are assumed to have a Gaussian probability distribution with standard deviation $(2k/3)^{1/2}$. The u_i' at any required location is then obtained by randomly sampling the distribution and changes discontinuously after each passage of the droplet-turbulence interaction time t_{int}. The time t_{int} corresponds physically to an eddy life time t_e or to a time t_{tr} for a particle to traverse an typical turbulent eddy. In [3,5,10,17] different formulations in choosing the t_e, t_{tr} and t_{int} are proposed. Here we choose $\ell_e = 1.65C_\mu^{3/4}k^{3/2}/\epsilon$ for the characteristic size of an eddy and $t_e = 3C_\mu k/\epsilon$ for the eddy lifetime to match the asymptotic dispersion analysis of Hinze [18].

The particle dispersion experimental setup of Snyder and Lumley [19] in a grid-generated turbulent flow was used for the numerical model validation. Particle densities and sizes are chosen to examine the phenomena in which the eddy lifetime controls interaction times (46.5 μm diameter hollow glass) , the transit time controls interaction times (87.0 μm corn), or the controlling-interaction times undergo transition from transit time to eddy life time (87.0 μm solid glass). In this experiment fluid turbulence intensities and length scale information were measured. The particle calculations were started at the experimental particle injection point of $x/m = 20$ (m is a 2.54-cm-square mesh). The particle velocity was assumed equal to the mean fluid velocity of 6.55 m/sec. 5,000 computational particles were sampled to calculate the resulting mean squared dispersion with respect to time.

Comparison of the predicted and measured particle dispersion is shown in Fig.1. The agreement is considered quite good. The comparison here is more favorable compared to the previous calculations by [17] and [5] especially for the medium particles for which the controlling particle /turbulence interaction time goes through a transition from t_e to t_{tr}. In this study, we did not estimated t_{int} as suggested by [17] in which t_{tr} was calculated from

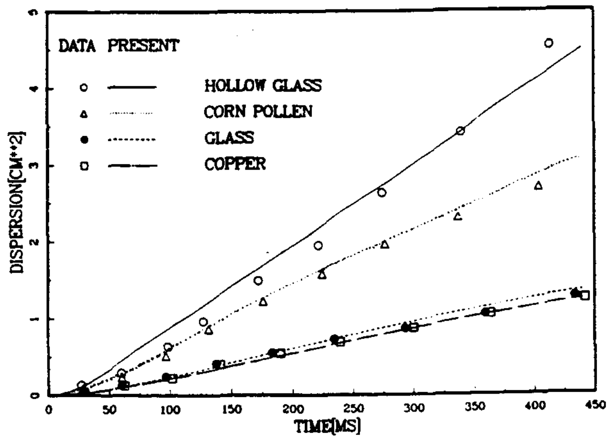


Figure 1. Particle Dispersion in a Grid-Generated Turbulent Flow.

a simplified B-B-O equation without the gravity effect. Instead, we follow the stochastic procedure suggested by Nichols [20] and trace particle trajectories as time progresses. This method has the flexibility of taking into account both the gravity effect (crossing trajectory effect) and the non-Stokian drag law and gives more statisfactory results for medium particles.

Single Fuel Injector Case

The experiments of Hiroyasu and Kadota [21] are used to validate the transient non-evaporating spray calculations. To rigorously model this flow, detailed atomization processes have to be resolved at the injector nozzle. Since this phenomena is not modeled in this study, information about the injected droplet size distribution, as well as the velocity distribution has to be estimated in vicinity of the injector at which the calculation starts. Following the suggestions of [10], the particle initial injecting velocity was determined by the mass flow rate and pressure difference of the nozzle. The estimated initial spray angle was also guided by [10]. The initial particle size distribution was given by the following form :

$$f_d(d_p) = \frac{6}{D_{32}} \exp(\frac{-3d_p}{D_{32}})$$

where D_{32} is the Sauter mean diameter.

The test conditions are given in Table 1. The spray was assumed to be axially symmetric, and the calculation was carried out in cylindrical coordinates. The computed penetration of the tip of the spray as a function of time for the two test conditions are qualitatively shown in Fig. 2 and quantitatively compared with the experimental data in Fig. 3. The comparison is very good here. Although the initial jet penetration depends greatly on the assumed initial spray condition, the consequent good prediction at the latter time demonstrates the accuracy of the numerical scheme and importance of the droplet-gas interaction.

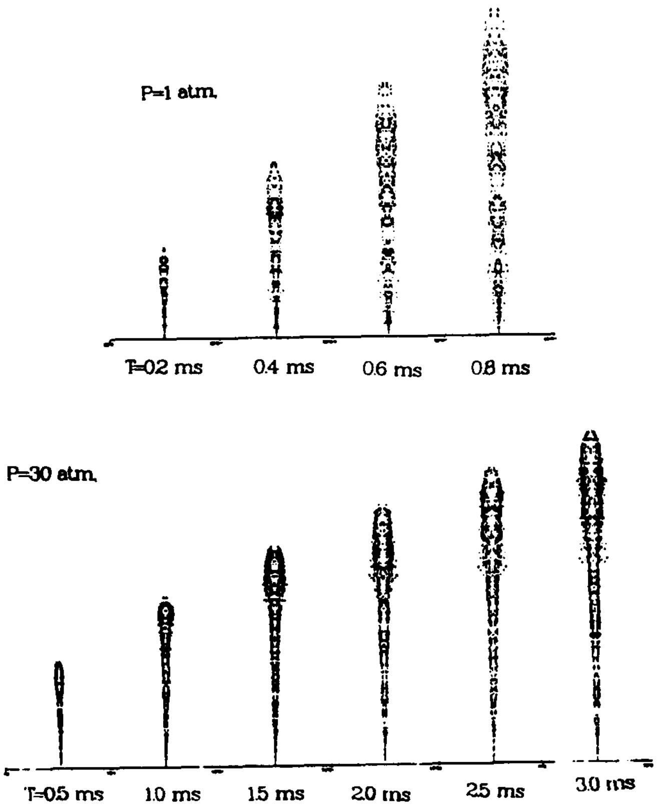


Figure 2. Particle Plots of a Single Orifice Spray.

TABLE 1

SINGLE-ORIFICE INJECTION PARAMETERS[21]

Chamber Gas Pressure (atm)	Injection Velocity (m/sec)	Gas Density (kg/m^3)	Mass Flow (kg/sec)	Sauter Mean Radius(SMR) μm	Nozzle Pressure Difference $\Delta P(atm)$
1	122:2	1.123	0.00726	5.0	98
30	102.5	33.70	0.00609	5.0	69

Fuel: Diesel fuel oil, $\rho_d =$ 840 kg/m^3

Ambient Gas: Nitrogen

Nozzle Diameter: 0.3 mm

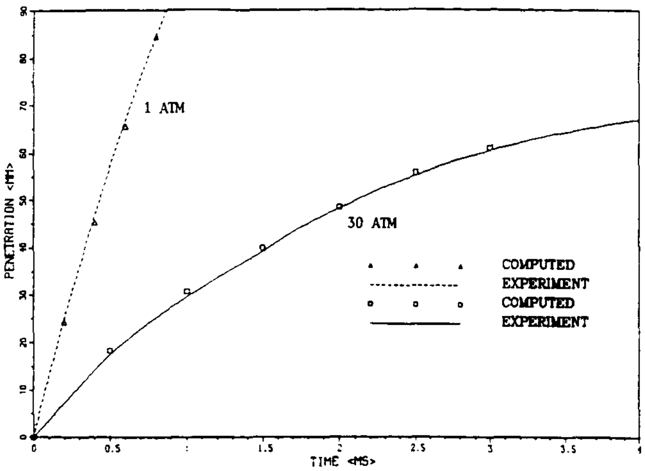


Figure 3. Comparison of Computed Spray Penetration with Experimental Data.

Hollow-Cone Spray

A polydispersed pulsed hollow-cone spray case of practical importance is also chosen for the test condition listed in Table 2. Fig. 4 shows the particle plot and the gas velocity vectors for a 30° spray. With the back pressure 1 atm, the interaction between the gas and droplets is seen to be rather strong. The shape of the spray is no longer conical even for very short time and the spray penetration is suppressed due to the interaction of the droplets with the induced air flow. The gas velocity vectors indicate the presence of a vortex near the head of the spray, which curls the spray tip toward the outside of spray. A substantial region of strong inward flow in the center of the cone near the injector was also observed. These flow patterns and spray shapes compared quite favorably with the experimental observations (c.f.[22]).

The efficiency assessment of the present numerical method was shown in Table 3 for the single-orifice and hollow-cone spray cases. The CPU times on a CRAY X/MP using the MAST code utilize the preset method and the TEACH code with PSIC method [3] for both transient spray calculations using $\Delta t = 0.1 ms$ are given. It can be seen that the amount of CPU-time is reduced about one order of magnitude using the present calculation procedure. Also, the present method is rather particle number independent. This is due to the fact that particles are injected at each time step and the source terms in the continuous phase are updated for all the particles at each Eulerian control volume. While in the TEACH/PSIC method, all the particles have to be tracked and the continuous phase flow field is held frozen between the global iterations at each time step. This PSIC algorithm thus requires substantial computer time and is inherently unsuitable for transient calculations.

TABLE 2

HOLLOW-CONE SPRAY PARAMETER

Chamber Gas Pressure (atm)	**Injection Velocity** (m/sec)	**Injection Angle** (**Degree**)	**Gas Density** (kg/m^3)	**Mass Flow** (kg/sec)	**Sauter Mean Radius(SMR)** μm
1	20.0	30	1.123	4×10^{-4}	2.5

TABLE 3

EFFICIENCY ASSESSMENT (CPU Time)

	MAST-2D		TEACH/PSIC	
SINGLE-ORIFICE SPRAY	Particle #		Particle #	
41 × 61 Grid	600	126.9 sec	800	1420 sec
300 Time Steps	1200	135.7 sec		
HOLLOW-CONE SPRAY	400	74.9 sec		
31 × 31 Grids			800	934 sec
200 Time Steps	1000	88.3 sec		

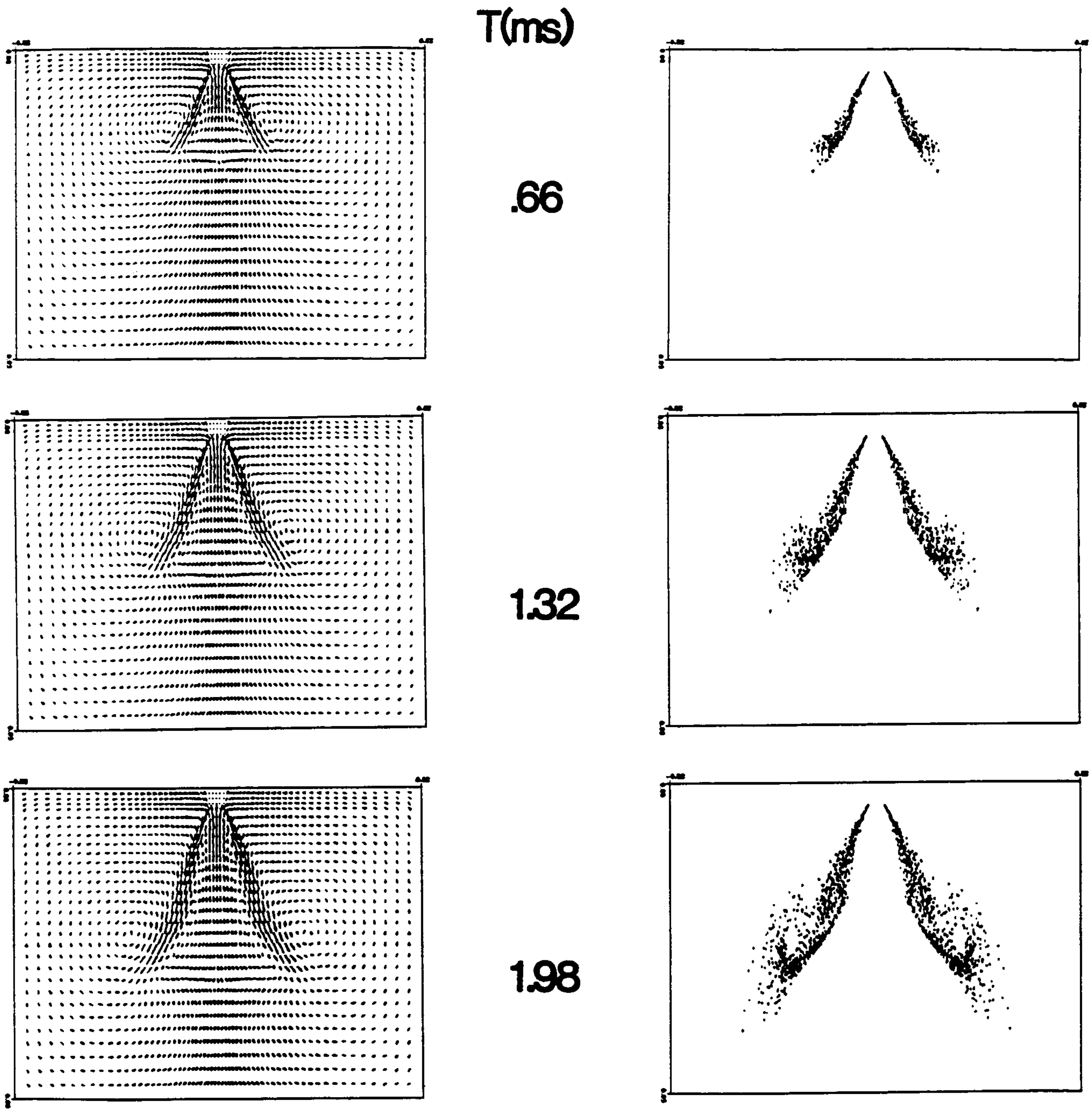


Figure 4. Velocity Vectors and Particle Plot of a 30° Hollow-cone Spray.

IV. SUMMARY

A new numerical scheme based on a non-iterative predictor-corrector pressure velocity coupling procedure has been developed for transient gas-droplet two-phase flows. The present scheme is formulated on a Eulerian - Lagrangian analysis and the two-way interaction between the two-phase is handled through a strong coupling procedure. This procedure eliminates global iterations conventionally used in the PSIC procedure and shows drastic saving in CPU time for transient spray calculations. This method has been extended to evaporating spray calculation recently [13]. Good agreements between the calculated results and the experimental data have been obtained despite the uncertainties in the inlet conditions for both fluid and droplets. The development of the technique is based on the assumption of dilute (non-iterating) sprays. The computer code, however, allows easy alteration of models, so that an appropriate model to suit the physical problem of interest can be quickly implemented. Experimental studies with better defined inlet conditions would be extremely useful in further model validations.

Acknowledgements

This study is supported by NASA grant (NAG8-092). The authors wish to acknowledge the CRAY CPU time supplied by the Alabama Supercomputer Network through UAH and the skillful typing of Miss C.M. Lin.

REFERENCES

1. Faeth, G.M., "Mixing, Transport and Combustion in Sprays", Prog. Energy Combust. Sci.,13, pp. 293-345, 1987.
2. See, Proc. Int. Conference on Mechanics of Two-Phase Flows, Eds. Richard S. L. Lee and F. Durst, Taipei, Taiwan, 1989.
3. A. Fashola, A. and Chen, C.P., "Modeling of Confined Turbulent Fluid -Particle Flows Using Eulerian and Lagrangian Schemes", Int. J. Heat and Mass Transfer, 33, pp. 691-700, 1990.
4. Mostafa, A.A. and Mongia, H.C., "On the Modeling of Turbulent Evaporating Sprays : Eulerian versus Lagrangian Approach", Int. J. Heat and Mass Transfer, 30, pp. 2583-2593, 1987.
5. Gosman, A.D. and Ioannides, E., "Aspects of Computer Simulation of Liquid-Fueled Combusters", AIAA paper 81-0323, 1981.
6. Crowe, C.T., Sharma, M.P., and Stock, D.E., "The Particle-Source- in Cell (PSI-Cell) Model for Gas-Droplet Flows", J. of Fluid Eng.,99, pp. 325-332, 1977.
7. Durst, F., Milojevic, D., and Schonung, B., "Eulerian and Lagrangian Predictions of Particulate Two-Phase Flows : A Numerical Study", Appl. Mech. Model, 8, pp. 101-115, 1984.
8. Ashiem, J.P. and Peters, J.E., "Alternative Fuel Spray Behavior", J. of Propulsion and Power, 5, pp. 391-398, 1989.
9. Boysan, J., Ayers, W.H., and Swithenbank, K., "A Fundamental Modeling Approach to Cyclone Design", Trans. IChemE, 60, pp. 222-230, 1982.
10. Dukowicz, J.K., "A Particle-Fluid Numerical Model for Liquid Sprays", J. Comp. Physics, 35, pp. 229-253, 1980.
11. Raju, M.S. and Sirignano, W.A., "Multicomponent Spray Computations in a Modified Centerbody Combuster", J. Propulsion and Piwer,6, pp. 97-105, 1990.
12. Sabnis, J.S. and de Jong, F.J., "Calculation of the Two-Phase Flow in an Evaporating Spray Using an Eulerian-Lagrangian Analysis", AIAA paper 90-0447, 1990.
13. Shang, H.M., Chen, C.P., and Jiang, Y., "Turbulence Modulation Effect on Evaporating Spray Characterization", AIAA paper 90-2442, 1990. Submitted to Int. J. Heat Mass Transfer.
14. Chen, C.P., Jiang, Y., and Shang, H.M., "MAST - a Computer Code for Multiphase All-Speed Transient Flows", NASA NAG8-092, 1990.
15. Jiang, Y., Chen, C.P., and Tucker, P.K., "Multigrid Solution of Unsteady Navier-Stokes Equations Using a Pressure Method", to appear in Numerical Heat Transfer J., 1991.
16. Issa, R.I., "Solutions of the Implicitly Discretized Fluid Flow Equations by Operator-Splitting", J. Comp. Physics, 62, pp. 40, 1985.
17. Shuen, J.S., Chen, L.D., and Faeth, G.M., "Evaluation of a Stochastic Model of Particle Dispersion in a Turbulent Round Jet", AIChE J., 29, pp. 167-170, 1983.
18. Hinze, J.O., Turbulence, 2nd Ed., Chapter 5, McGraw-Hill, 1975.
19. Snyder, W.H. and Lumley, J.L., "Some Measurements of Particle Velocity Autocorrelation Functions in a Turbulent Flow", J. Fluid Mech., 48, pp. 41-71, 1971.
20. Nichols, R.H., "The Effect of Particle Dynamics on Turbulence Measurements with the Laser Velocimeter", in Numerical Methods for Multiphase Flows, ASME FED-Vol. 91, pp. 35-45, 1990.
21. Hiroyasu, H. and Kadota, T., "Fuel Droplet Size Distribution in Diesel Combustion Chamber", SAE paper 740715, 1974.
22. Meintjes, K., "Engine Combustion Modeling : Prospects and Challenges", CRAY CHANNELS, Winter issue, pp. 12-15, 1987.

REFERENCES

1. C. M. Rhie, " A Pressure Based Navier-Stokes Solver Using the Multigrid Method", AIAA paper, 86-0207, 1986.
2. W.Shyy, S.S. Tong and S. M. Correa, " Numerical Recirculating Flow Calculation Using a Body-fitted coordinate System," Numerical Heat Transfer, 8, pp 99-113, 1985.
3. C. R. Maliska , G. D. Raithby, " A Method for Computing 3-D Flows Using Non-Orthogonal BFC, " Int. J. Numerical Method Fluids, 4, pp. 519-537, 1984.
4. K. C. Karki and S. V. Patankar, " Pressure Based Calculation Procedure for Viscous Flows at All Speeds in Arbitrary Configurations," AIAA J. 27, pp. 1167-1174, 1989.
5. REFLEQS-2D: A Computer Program For Reactive Flows, CFDRC, GR-88-4, 1988.
6. D. A. Anderson, J. C. Tannehill and R. H. Pletcher, Computational Fluid Mechanics and Heat Transfer, Hemisphere Pub. N.Y. 1984.
7. S.V. Patankar, Numerical Heat Transfer and Fluid Flows, McGraw-Hill, 1980.
8. S. R. Chakravarthy and S. Osher, " A New Class of High Accuray TVD Schemes for Hyperbolic Conservation Laws" , AIAA paper, 85-0363,1985.
9. J. P. van Doormaal, G.D. Raithby and B. H. McDonald," The segregated Approach to Predicting Viscous Compressible Fluid Flows, " J. F. Turbomachinery, ASME, 109, pp. 268-277, 1987.
10. R. I. Issa, "Solutions of the Implicitly Discretized Fluid Flow Equations by Operator Splitting," J. Comp phys. 62, 40-65, 1985.
11. Y. Jiang, " Studies of Pressure-Velocity Coupling for Fluid Flows at All Speeds," Ph.D. Thesis, The University of Alabama in Huntsville , AL, 1991.
12. C. M. Rhie and W. L. Chou, "Numerical Study of the Turbulent Flow Past and Airfoil with Trailing Edge Separation," AIAA J. 12, pp,1525-1532, 1983.
13. M. Peric, R. Kessler and G. Scheuer," Comparison of Finite Volume Numerical Methods with Staggered and Collocated Grids," Computer and Fluids, 16, pp.389-398, 1988.
14. D. Howard, W. M. Connoley and J. S. Rollett," Unsymmetric Conjugate Gradient Methods and Sparse Direct Methods in Finite Element Flow Simulation," Int. J. Numerical Method Fluids, 10, pp.925-945, 1990.
15. C. P. Chen, H. M. Shang and Y. Jiang, "A Novel Gas-Droplet Numerical Method for Spray Combustion, " AIAA paper, 91-0286, to appear, Int. J. Numerical Meth. Fluids, 1992.

16. A. A. Amsden, P. J. O'Rourke, T. D. Butter, "KIVA-II, A computer Program for Chemically Reaction Flows with Sprays, " LA-11560-MS, UC-9b, 1989.

17. S. Gordon and B. J. McBride, " Computer Program for calculation of Complex Chemical Equilibrium," NASA SP-273, 1971, 1976.

18. D. R. Stull and H. Prophet, " FANAF Thermochemical Tables, " 2nd ed. NSRDS-SBS 37, 1971. N. W. Chase et al. J. phys. Chem. Ref. Data 3, 311,1974.

19. B. F. Armaly, F. Durst, J.C.F. Pereira and B. Schonung, " Experimental and Theoretical Investigation of Backward-Facing Step Flow," J. Fluid Mech. 127, pp.473-496, 1983.

20. W. P. Jones, "Models for Turbulent Flows with Variable Density and Combustion, " in Prediction Method for Turbulent Flows, pp.379-422, Ed. W. Kollman, Hemisphere, 1980.

21. Y. M. Kim and T. J. Chung, "Turbulent Combustion Analysis with Various Probability Density Functions, " AIAA paper 89-1991, 1989.

22. Y. Jiang, C. P. Chen and P. K. Tucker, "Multigrid Solution of Unsteady Navier-Stokes Equation Using a Pressure Method," Numerical Heat Transfer, Part A, 20, pp.81-93, 1991.

23. R. H. Ni, "A Multiple Grid Scheme for Solving the Euler Equations," AIAA J. 20, pp.1565-1571, 1982.

24. C. M. Rhie, "A Pressure Based Navier-Stokes Solver Using the Multigrid Method," AIAA paper 8b-0207, 1986.

25. S. Eidelman, P. Colella and R. P. Shreeve, "Application of the Godunov Method and Its Second-Order Extension to Cascade Flow Modeling," AIAA J. 22, 1609-1615, 1984.

26. S. Parameswaran and R. Sun, "Numerical Aerodynamics Simulation of Turbulent Flows Around a Car-like Body Using the Non-Staggered Grid System," AIAA 89-1884, 1989.

27. H. Hiroyasu and T. Kadato, "Fuel Droplet Size Distribution in Diesel Combustion Chamber," SAE Paper 740715, 1974.

28. Y. M. Kim, H. M. Shang, C. P. Chen, and Wang, T. S., "Characterization andModeling for Evaporating and Combusting Fuel Sprays," to be appeared in the proceedings of the Second National Fluid Dynamics Congress, June 22-25, 1992, Los Angeles, CA.

29. C. P. Chen, H M. Shang, Y. Jiang (1990), "A Novel Gas-Droplet Numerical Method for Spray Combustion", Eigth Liquid Rocket Engine CFD Working Group Meeting, NASA-MSFC, April 17-19, 1990, Submitted to J. Comput. Physics.

30. A. J. Shearer and E. G. Groff, "Injection System Effects on Oscillating-Poppet-Injector Sprays," Proceedings of the ASME Diesel and Gas Engine Power Division Conference, New York, pp.33-42, 1984.

31. K. H. Khalil, F. M. El-Mahallawy, H. A. Moneib, "Effect of Combustion Air Swirl on the Flow Pattern in a Cylindrical Oil Fired Furnace", Sixteenth Symposium on Combustion, The Combustion Institute, Pittsburgh, PA, pp. 135-143, 1977.

32. Y. M. Kim, H. M. Shang, c. P. Chen, "Non-Isotropic Turbulence Effects on Spray Combustion", AIAA Paper 91-2196, 27th Joint Propulsion Conference, Sacramento, CA, June 24-26, 1991.

33. P. J. O'Rourke "Collective Drop Effects on Vaporizing Liquid Sprays", Los Alamos National Laboratory report LA-9069-T, 1981.

34. P. J. O'Rourke 'The TAB Method for Numerical Calculation of Spray Droplet Breakup", SAE Technical Paper 872089,1987.